REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
09-07-2003	Final Report	28 May 2002 - 28-May-03

4. TITLE AND SUBTITLE

Cooperative Control of Autonomous Flight Vehicles Formation

5a. CONTRACT NUMBER
F61775-02-WE031

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Professor Mario Innocenti

5d. PROJECT NUMBER

5d. TASK NUMBER

5e. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Universita' di Pisa
Via Diotisalvi 2
Pisa 56126
Italy

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

EOARD
PSC 802 BOX 14
FPO 09499-0014

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
SPC 02-4031

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

20040715 174

14. ABSTRACT

This report results from a contract tasking Universita' di Pisa as follows: Research will deal with aspects of automated formation management and control necessary for development of a dual controller capable of keeping a given formation structure and allowing the formation to follow specified paths. The report contains two parts: Part 1 describes the results obtained and methodologies develop for dynamics and control of unmanned air vehicles flying in a formation. The second part describes a fuzzy-set approach to the guidance and control management of vehicles in order to establish conditions for successful target intercept.

15. SUBJECT TERMS
EOARD, Communications, Control, Guidance

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18, NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			NEAL D. GLASSMAN
UNCLAS	UNCLAS	UNCLAS	UL	136	19b. TELEPHONE NUMBER *(Include area code)* +1 703-696-9548

COOPERATIVE CONTROL OF AUTONOMOUS FLIGHT VEHICLES FORMATION

Mario Innocenti

Department of Electrical Systems and Automation

University of Pisa, 56126 Pisa, Italy

Final Report

EOARD Contract F61775-02-WE031

29th March 2004

Abstract

This report describes the results obtained for the contract EOARD F61775-02-WE031, during the period June 2002-May 2003. The research was performed at the University of Pisa, Italy, in the Department of Electrical Systems and Automation, with Prof. Mario Innocenti as principal investigator. The personnel involved in the work included also Prof. Andrea Caiti, Dr. Lorenzo Pollini, and Mr. demetrio Turra, respectively associate professor, post-doctoral fellow, and Ph.D. student in the same department. Dr. Fabrizio Giulietti, currently post-doctoral fellow at the University of Bologna (Forlì campus) participated to the effort primarily on the aspects related to formation dynamics and control. The contractor's technical monitor was Dr. Neal Glassman from EOARD, and the contact point for the actual detailed aspects of the work was Mr. Fred Davis (AFRL/MNA). The report contains two parts: Part 1 describes the results obtained, and methodologies developed, for dynamics and control of unmanned air vehicles flying in a formation. The second part describes a fuzzy-set approach to the guidance and control management of vehicles, in order to establish conditions for succesfull terget intercept. From the research activity performed within the present effort, the following publications were submitted/accepted/published:

- Innocenti M., Pollini L., Turra D., "A Fuzzy Approach to the Guidance of Unmanned Air Vehicles Tracking moving Targets", submitted to the IEEE Transcactions on Control Systems Technology, June 2003.

- Pollini, L., Giulietti, F., Innocenti, M., "Dynamic and Control Issues of Formation Flight", Aerospace Science and Technology, submitted June, 2002.

- Innocenti M., Pollini L., Turra D., "A Guidance System for Unmanned Air Vehicles based on fuzzy Sets and fixed Waypoints", submitted to the AIAA Journal of Guidance, Control, and Dynamics, May 2003 (accepted, in press).

- Innocenti M., Pollini L., Marullo A., "Gain-Scheduling Stability Issues using Differential Inclusion and Fuzzy Systems", submitted to the AIAA Journal of Guidance, Control, and Dynamics, May 2003. (accepted, in press).

- Innocenti M., Pollini L, Giulietti F., "Management of Communication Failures in Formation Flight", submitted to the AIAA Journal of Aerospace Computing, Information, and Communication, Vol. 1, No. 1, pp.19-35, January 2004.

- Turra, D., Pollini, L., Innocenti, M., "Moving Waypoint-based Fuzzy Guidance for Unmanned Aircraft", AIAA Guidance, Navigation, and Control Conference, GNC03, Austin, TX, August 2003.

- Pollini, L., Mati, R, Innocenti, M., G.Campa, G., M.Napolitano, M., " A synthetic environment for simulation of vision-based formation flight,", AIAA Modeling and Simulation Technologies, MST2003, Austin, TX, August 2003.

- Pollini, L.,Baralli, F., Innocenti, M., "Waypoint-based Fuzzy Guidance for Unmanned Aircraft - A New Approach", AIAA Guidance, Navigation and Control Conference, Monterey, California, August 2002.

- Giulietti, F., Napolitano, M., Capetta, R., Innocenti, M., "Detailed Modeling of Multiple Aircraft within close Formation", AIAA Atmospheric Flight Mechanics Conference, Monterey, California, August 2002.

Contents

List of Figures

Part I

Dynamics and Control of Formation Fligh

Chapter 1

Formation Flight Dynamic Modelling

1.1 Introduction

The quest for increased performance in unmanned aerospace vehicles yields new aerodynamic configurations mainly tailless (stealth capabilities), with high angles of attack for post-stall aerodynamic regimes (maneuverability), including thrust vectoring capabilities (agility, performance, and survivability following battle damage). In addiction, the use of multi-aircraft formations is becoming of interest for a variety of missions. The present chapter is related to the latter aspect, that is formation flying of Unmanned Air Vehicles (UAVs). Until recently, UAVs have been primarily used as test bed. For instance they were used to test flight envelope expansion and flight control systems for the manned aircraft or as a target for weapon systems. Over the last ten years, UAVs have been thought as operational airborne platforms in their own rights, and have received considerable attention especially in military applica-

tions. Today we foresee with new aerospace scenarios, where UAVs can take many roles previously performed by manned aircraft, as well as new tasks that are more suited to unmanned vehicles, as surveillance, reconnaissance and rescue in hostile environment.

The operational potential of UAVs could strongly improved by making them flying within a close formation. The first advantage comes from aerodynamic effects. It is well known in fact that aircraft with large aspect ratio wings have better overall aerodynamic efficiency because of reduction in drag for a given lift. However, large aspect ratio implies large wingspan for a given area, this means that the resulting structure will be unreasonably flexible and fragile for lightweight design . A similar improvement in global efficiency can be achieved by flying multiple aircraft in close formation. In an idealized case of n identical aircraft, each with the aspect ratio AR, flying in tip to tip formation, the effect would be that of a single aircraft with $n \times AR$ aspect ratio. The aerodynamic benefits are due to favorable wake-vortex encounters. Wind tunnel tests and analytical studies have shown that the benefits increase as additional aircraft are added to the formation. Moreover, from an operational point of view, many aircraft involved in a mission can be better managed if they fly in a formation, rather than in an undefined structure.

The first, important challenge in the study of formation flight is represented by the complexity of the aerodynamic coupling. The mathematical modeling of the aerodynamic interference between different aircraft in a formation was first approached by Bloy and others [Bloy, A. W. and Jouma'a, M., 1995], [Bloy, A. W. *et al.*, 1993]. They considered the problem of aerodynamic interference on lateral directional stability during air-to-air refuelling maneuvers. In Reference [Myatt, J. H. and Blake, W., 1999], Myatt and Blake proposed an aerodynamic database with experimental data for the

follower aircraft in a two-vehicle formation. In a more recent work, Gingras and co-workers [Gingras, D., 1999], [Gingras, D. and Player, J., 2001], proposed wind tunnel testing to acquire data for close vehicles simulation, while Nelson and Jumper used the Horse-shoe Vortex theory to discuss the response of an airplane following an encounter with the trailing vortex wake by another airplane, [Nelson, R. C., 2001].

The problem of aerodynamic interference in a close formation flight was further described by Blake, D'Azzo and Multhopp, with a Leader/Wingman structure [Proud. A. W. *et al.*, 1999], [Blake, W. and Multhopp, D., 1998]. The Leader generates vortices behind its wing. Such vortices exert actions on the Wingman lifting surfaces. In their approach, the Leader's wake is modeled via Horse-shoe Vortex theory [Houghton, E.L. and Brock, A.E., 1970], and the drag reduction, sideslip force and induced angle of attack are introduced in the Wingman dynamics through additional stability derivatives leading to a mathematical aircraft model for a formation flight.

Close formation flight control, intended as a guidance, navigation and control problem, was originally studied for a classic leader/wingman configuration. An aircraft (Leader) is selected to direct the formation, following a prescribed path, and all the other airplanes (Wingmen) are expected to maintain a fixed relative distance with respect to the lead airplane, in order for the formation to maintain a desired geometrical shape. D'Azzo and co-workers analyzed the kinematic coupling effect of the two-aircraft Leader/Wingman configuration, and introduced a proportional integral (PI) controller for formation control [Buzogany, L. E. *et al.*, 1993]. The concept of decentralized control for UAVs in formation flight was first introduced by Wolfe, Chichka, and Speyer [Chichka, D. and Speyer, J., 1996]. The advantages of using decentralized controllers are clear, especially for a large size UAVs formation, where

a full state feedback solution for the problem could be quite unfeasible due to the very high number of states. In addition, a decentralized control approach would provide more flexibility for a varying number of UAVs in the formation. Later work by Chichka, Speyer [Chichka, D. and Speyer, J., 1996], Chichka, Speyer, and others [D. F. Chichka *et al.*, 1999] focused on a peak-seeking control approach for close formation flight, where the overall formation control was approached as an optimal control problem with the objective of minimizing drag. A fairly intensive and inclusive study on formation flight control - with the use of a PI control scheme - was presented by Proud, Pachter, and D'Azzo for a 2-D formulation of the problem [M. Pachter and J.L. Dargan, 1994], and then recently exteded to a 3-D formation dynamics problem [Proud. A. W. *et al.*, 1999], [M. Pachter *et al.*, 2001]. Within the area of formation control, adaptive control approaches were recently presented by Schumacher and Kumar [Schumacher and Kumar, 2000]. This adaptive scheme featured an optimal LQR design for the outer loop and a Dynamic Inversion design for the inner loop. In reference [F. Giulietti *et al.*, 2000], two different Leader-Wingman structures were developed. In Leader-Mode, each Wingman takes the trajectory reference from the Leader of formation while in Front-Mode each aircraft takes its reference from the preceding one.

The chapter is organized as follows. Section 2 describes aerodynamic modeling issues, sections 3 and 4 describe the formation modeling, and the three dimensional code used for aerodynamic interference calculation is presented. Given a lifting surface system, the proposed technique permits the calculation of induced velocities, and the forces due to coupling effects between aerodynamic surfaces. Thus, the above results are used to obtain the dynamic model of an aircraft flying within a formation. Moreover the kinematic equations of the relative distance between aircraft are de-

veloped. Section 5 presents a different approach to formation flight. In the proposed strategy, each aircraft does not refer to the preceding one or to the formation leader, but keeps its position with respect to an imaginary point in the formation whose dynamics depend on all the aircraft positions. The approach is based on the apparent behavior of some migrational birds, that during a migrational flight 'wait' for the those birds which have changed the original geometry of the formation by flying in a different path. The formation controller is constituted by two components: a trajectory controller, which provides tracking of a prescribed path, and a position controller which permits formation geometry keeping. These control laws are mixed by a parameter that depends on the position error. Simulations showing the application of the outlined control laws are presented, including a comparison between the classical Leader/Wingman structure and the proposed strategy.

1.2 Aerodynamic Modeling

This section describes the method used to calculate the aerodynamic coupling effects introduced by close formation flight on each aircraft within the formation.

The single Horse-shoe Vortex technique consists in replacing the whole lifting surface by a single vortex made with two free segments and one wing-fixed segment. Such model is simple and introduces minor modifications to the dynamics equations. However, by modeling a close formation by the single Horse-Shoe Vortex theory, only one aircraft is affected by the wake efforts and the effect of the wake along the x-axis is not modeled in order to obtain a close analytical solution. This may cause the results not to be very accurate and complete. In fact, in a formation flight problem, the mathematical modeling for all the aircraft involved in the formation is a critical

issue, especially since the experimental data from wind tunnel tests may be too costly in case of an increasing number of vehicles.

In this paper, a different 3-D approach based on a wing-distributed Horse-shoe vortex system, is used. Given a system of lifting surfaces, this technique features the calculation of induced velocities as well as the forces and moments due to the coupling effects between all the aerodynamics surfaces. The proposed methodology has been coded within WakeCAD, a Matlab detailed code designed for formation flight modeling issues [Capetta, R. *et al.*, 2001]. Within the approach, wings are not modeled with a single Horse-shoe Vortex; instead all the lifting surfaces are modeled and they are represented by a 'vortex system' whose dimensions can be set on the base of the required accuracy. Furthermore, the induced velocities relative to all lifting surfaces are calculated. WakeCAD outputs are induced velocities lift distributions, forces coefficients as well as the entire set of stability and control derivatives.

1.2.1 The Distributed Horse-Shoe Vortex Technique

The first step consists in defining the lifting surfaces. Only single tapered wings are considered within this effort. To define each lifting surface, the following vector of geometrical and aerodynamic parameters of the surface profile is introduced:

$$R_p = \begin{bmatrix} \delta_h & \Lambda & \sigma & c_r & c_t & C_{L_\alpha} & C_{m_0} & \alpha_\infty \end{bmatrix} \tag{1.1}$$

where δ_h, Λ, and σ are the dihedral, swept and twist angle of the lifting surface respectively, c_r and c_t are the root and tip geometric chord, while C_{L_α}, C_{m_0} and α_∞ are lift and moment derivatives and the angle of attack of the root profile. Intermediate profiles parameters are calculated by a linear interpolation from the preceding ones.

Once a number s of lifting surfaces has been defined, each surface is modeled

through a straight line, also known as *lifting line*, passing through the aerodynamic center of the local profile. The aerodynamic center of the single profile is approximately set at 26% of the geometric chord. Let us now consider the $r-th$ lifting surface. First the two halves of its lifting line are divided in $n_r/2$ segments. Then a vortex system, composed by a number n_r of horse-shoe vortices, is distributed on each lifting line. Each horse-shoe vortex is modeled with a wing-fixed portion and two free portions that extend to infinity. Thus the two following subsets of points are introduced:

$$P_r = \{ P_{k_r} \in \mathbb{R}^3;\, k = 1 \ldots n_r + 1 \} \tag{1.2}$$

and

$$m_r = \{ m_{k_r} \in \mathbb{R}^3;\, k = 1 \ldots n_r \} \tag{1.3}$$

The first subset, P_r, contains the points where the free portions of each horse-shoe vortex leaves the lifting line. For example, given the $k_r - th$ vortex, the points P_{k_r} and P_{k_r+1} are the points where the free portions of this vortex leave the lifting line. The second subset, m_r, contains the points where the induced velocity is evaluated. They are the middle points of the wing-fixed portions of each horse-shoe vortex. The point m_{k_r} is the middle point of the segment $P_{k_r}P_{k_r+1}$. Figure 1.1 shows, as an example, the generic $r-th$ lifting surface with the vortex system along with the points P_{k_r} and m_{k_r}.

1.2.2 Induced Velocity Calculation

Once the lifting surfaces have been defined, in terms of lifting line and vortex systems, the induced velocity, and the strength of the vortices may be evaluated. The aerodynamic interference is calculated by locally using the fluid dynamic analogue of

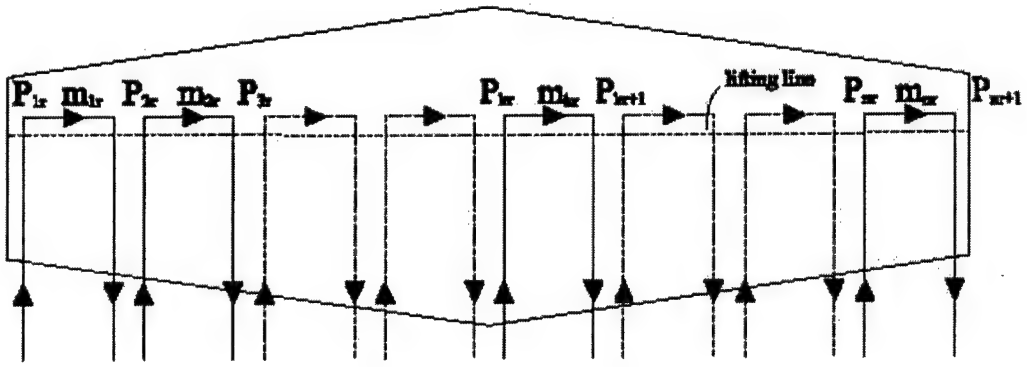


Figure 1.1: Lifting Surface and Vortex System

the Biot-Savart law from electromagnetics. Such law is applied both to the wing-fixed portion and to the free portions of each vortex.

Consider the generic vortex segment defined between the points P_1, P_2. The velocity induced by this vortex segment on the point m is orthogonal to the plane containing the vortex segment and the vector P_1m, as shown in Figure 1.2. The induced velocity direction is then given by:

$$\nu_{(d_{12},m)} = \frac{d_{12} \times P_1 m}{\mid d_{12} \times P_1 m \mid} \tag{1.4}$$

where the vector d_{12} is the vector associated with the vortex segment. Let us now introduce the induction function $\Phi_{(d_{12},m)}$ for the given vortex segment, defined as the induced velocity per unit value of vortex strength. Its expression, according to the Biot-Savart law, is given by:

$$\Phi_{(d_{12},m)} = \frac{1}{4\pi h_{(d_{12},m)}} \left[\cos\beta'_{(d_{12},m)} + \cos\beta''_{(d_{12},m)}\right] \cdot \nu_{(d_{12},m)} \tag{1.5}$$

where $h_{(d_{12},m)}$ is the distance between the point m and the segment d_{12}, while $\beta'_{(d_{12},m)}$

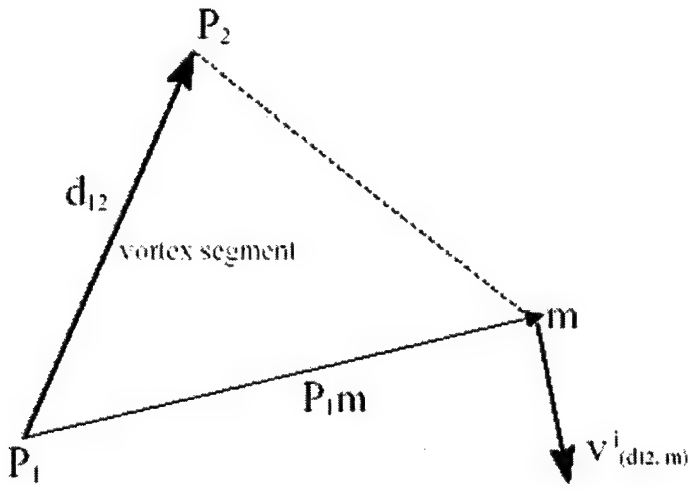


Figure 1.2: Induced Velocity Direction

and $\beta''_{(d_{12},m)}$ are the angles between d_{12} and the vectors P_1m and P_2m respectively. These quantities could be obtain as as described below. From Figure 1.3, it can be seen that:

$$d_{12} \cdot P_1m = \mid d_{12} \mid \cdot \mid P_1m \mid \cos\beta'_{d_{12},m} \tag{1.6}$$

and

$$b_{(d_{12},m)} = \mid P_1m \mid \cos\beta'_{(d_{12},m)} \tag{1.7}$$

by substituting equation (1.7) into (1.6), solving for $b_{(d_{12},m)}$, yields:

$$b_{(d_{12},m)} = \frac{d_{12} \cdot P_1m}{\mid d_{12} \mid} \tag{1.8}$$

With the distance $h_{(d_{12},m)}$ given by:

$$h_{(d_{12},m)} = \sqrt{\mid P_1m \mid^2 - b^2_{(d_{12},m)}} \tag{1.9}$$

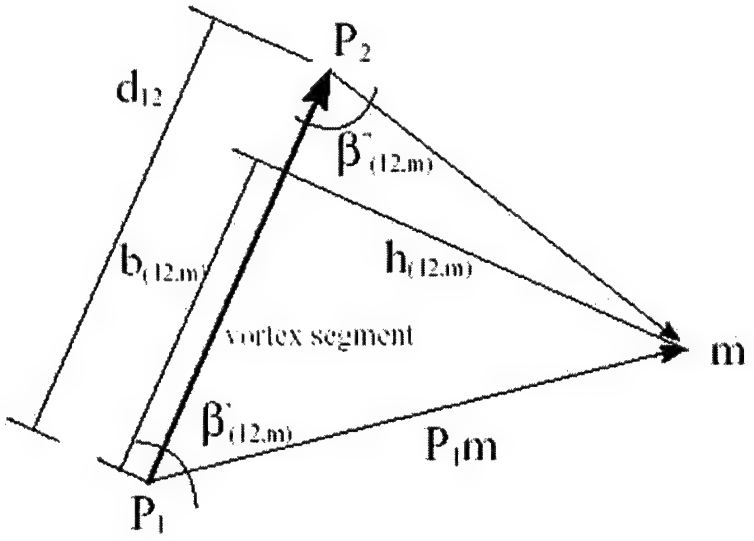


Figure 1.3: Induced Velocity Modulus Calculation

Finally, the quantities $\cos\beta'_{(d_{12},m)}$ and $\cos\beta''_{(d_{12},m)}$ are computed as:

$$\cos\beta'_{(d_{12},m)} = \frac{b_{(d_{12},m)}}{\mid P_1 m \mid} \tag{1.10}$$

and

$$\cos\beta''_{(d_{12},m)} = \frac{\mid d_{12} \mid - b_{(d_{12},m)}}{\mid P_2 m \mid} \tag{1.11}$$

If the vortex segment extends to infinity, that is P_1 or P_2 go to infinity, the values of $\beta'_{(d_{12},m)}$ or $\beta''_{(d_{12},m)}$ tend to zero respectively.

Now the induction function is defined, the induced velocity vector can be introduced as follows. Consider two generic aerodynamic surfaces, and let Γ_{k_r} be the strength of the $k-th$ horse-shoes vortex belonging to the $r-th$ surface. The velocity

induced by such vortex on the point m_{j_q} is given by:

$$V^i_{(k_r,j_q)} = \Gamma_{k_r} \cdot \Phi_{(k_r,j_q)} \tag{1.12}$$

It is important to notice that the induction function $\Phi_{(k_r,j_q)}$, is the sum of the contributions of the wing-fixed vortex segment and two free vortex segment. Then the following equation is imposed on the point m_{j_q}:

$$\rho\Gamma_{j_q} V_\infty = \frac{1}{2}\rho V_\infty^2 C_{L\alpha j_q} \left(\alpha_{j_q} - \alpha^i{}_{j_q}\right) c_{j_q} \tag{1.13}$$

Where $\alpha^i_{j_q}$ is the induced angle of attack on the point m_{j_q}. It is given by:

$$\alpha^i{}_{j_q} = \frac{V^i_{z(k_r,j_q)}}{V_\infty} = \frac{1}{V_\infty}\sum_{r=1}^{s}\sum_{k=1}^{n_r}\Gamma_{k_r}\Phi_{z(k_r,j_q)} \tag{1.14}$$

The substitution of $\alpha^i_{j_q}$ in equation (1.13) leads to:

$$\rho\,\Gamma_{j_q} V_\infty = \frac{1}{2}\rho V_\infty^2 C_{L\alpha j_q}\left(\alpha_{j_q} - \frac{1}{V_\infty}\sum_{r=1}^{s}\sum_{k=1}^{n_r}\Gamma_{k_r}\Phi_{z(k_r,j_q)}\right) c_{j_q} \tag{1.15}$$

where $\Phi_{z(k_r,j_q)}$ is the component along the z-axis of the induction function. The previous equation can be rewritten removing the dependency from the airspeed V_∞, by introducing a new variable:

$$\gamma_{j_q} = \frac{\Gamma_{j_q}}{V_\infty} \tag{1.16}$$

Thus equation (1.15) becomes:

$$\frac{\gamma_{j_q}}{2\,C_{L\alpha j_q}\,c_{j_q}} = \alpha_{j_q} - \sum_{r=1}^{s}\sum_{k=1}^{n_r}\gamma_{k_r}\Phi_{z(k_r,j_q)} \tag{1.17}$$

Furthermore, by imposing the previous equation (1.17) on each point $m_{j_q} \in m_q\,(q = 1\ldots s)$, a linear system in the standard form is obtained.

$$Ax = B \tag{1.18}$$

In this case matrix $A \in \mathbb{R}^{(N \times N)}$, where N is the dimension of the whole vortex system, is given by:

$$A = \begin{bmatrix} A_{11} & \dots & A_{1s} \\ \vdots & \ddots & \vdots \\ A_{s1} & \dots & A_{ss} \end{bmatrix} \tag{1.19}$$

Partition $A_{rq} \in \mathbb{R}^{(n_n \times n_q)}$ contains the induction coefficients in the points belonging to m_q, due to the vortices of the $r-th$ lifting surface. Such coefficient are given by:

$$A_{rq}(k,j) = \begin{cases} -\Phi_{z(k_r,j_q)} & r \neq q \\ \\ 1/2\, C_{L\alpha j_q}\, c_{j_q} & r = q,\ k = j \end{cases} \tag{1.20}$$

The matrix $B \in \mathbb{R}^{(N \times 1)}$ is given by:

$$B = \left[B_1^t,\, B_2^t,\, \dots,\, B_q^t,\, \dots,\, B_s^t\right]^t \tag{1.21}$$

where the partition $B_q \in \mathbb{R}^{(n_q \times 1)}$ contains angle of attack values in the points belonging to m_q:

$$B_q(j) = \alpha_{j_q} \tag{1.22}$$

The solution of the linear system provides the strength of the vortex system.

1.2.3 Force and Moment Coefficients

Once the value of the magnitude of the vortex system is known, aerodynamic forces, moments and coefficients can be finally computed. Consider the point m_{j_q}; the force acting on m_{j_q} is directed along a direction normal to the velocity V. Its magnitude is given by:

$$\mid F_{j_q} \mid = \rho\, V\, \Gamma_{j_q} = \rho\, V^2\, \gamma_{j_q} = \frac{1}{2}\, \rho\, V_\infty^2\, C_{F\, j_q}\, c_{j_q} \tag{1.23}$$

The velocity vector V is given by the sum of asymptotic velocity V_∞, and induced velocity V^i. From equation (1.23):

$$| \bar{F}_{j_q} | = C_{F j_q} c_{j_q} = 2 \gamma_{j_q} \tag{1.24}$$

where $\bar{F}_{j_q}$ is defined as the force for unit of dynamic pressure. By dividing the total force by the quantity S_q, the aerodynamic coefficients of the $q-th$ lifting surface are obtained using:

$$C_{F_q} = \begin{pmatrix} C_{x_q} \\ C_{y_q} \\ C_{z_q} \end{pmatrix} = \frac{2 \bar{F}_q}{S_q} \tag{1.25}$$

Similarly the aerodynamic moment coefficients, around point m_{j_q} can be obtained as follows. Consider the moment for unit of dynamic pressure:

$$\bar{M}_{j_q} = C_{M_0 j_q} c_{j_q}^2 + QP_{j_q} \times \bar{F}_{j_q} \tag{1.26}$$

where $C_{M_0 j_q}$ is a vector containing the moment coefficients of the profile in the point m_{j_q}, and QP_{j_q} is the distance vector between such point and a reference point Q. The total moment acting on the $q-th$ lifting surface is then given by:

$$\bar{M}_q = \sum_{j=1}^{n_q} \bar{M}_{j_q} \tag{1.27}$$

and thus the coefficients are expressed by:

$$C_{M_q} = \begin{pmatrix} C_{l_q} \\ C_{m_q} \\ C_{n_q} \end{pmatrix} = \frac{2 \bar{M}_q}{S_q l^\star} \tag{1.28}$$

where $l^\star$ is the mean aerodynamic cord in case of pitching moment and the wing span in case of rolling and yawing moment. Finally, force and moment coefficients

for each aircraft in the formation are calculated. In case of conventional aircraft, made composed with a main wing, an horizontal tail and a vertical tail, within an n_a-aircraft formation:

$$C_{F_1} = \sum_{t=1}^{3} C_{F_t}, \ \dots, C_{F_{n_a}} = \sum_{t=s-2}^{s} C_{F_t} \tag{1.29}$$

and

$$C_{M_1} = \sum_{t=1}^{3} C_{M_t}, \ \dots, C_{M_{n_a}} = \sum_{t=s-2}^{s} C_{M_t} \tag{1.30}$$

1.3 Formation Dynamics

This section describes the mathematical model of an aircraft flying within a formation. The main difference with the model of an 'isolated' aircraft consists in the value of the aerodynamic coefficients. The modeling procedures outlined in the previous section are introduced. Once the mathematical model is developed, the kinematic equations describing the relative distances between the aircraft are derived. These equations are the basis of formation control system design.

1.3.1 Modified Equations of Motion

The aerodynamic force and moment coefficients, evaluated in the previous section, can be directly introduced in the flight dynamic equations. However, it is useful to split the aerodynamic coefficients according two different contributions:

$$\begin{pmatrix} \boldsymbol{C}_F \\ \boldsymbol{C}_M \end{pmatrix} = \begin{pmatrix} \boldsymbol{C}_{Fi} \\ \boldsymbol{C}_{Mi} \end{pmatrix} + \begin{pmatrix} \boldsymbol{C}_{Ff} \\ \boldsymbol{C}_{Mf} \end{pmatrix} \tag{1.31}$$

The first term of the right hand side is the aerodynamic coefficient vector in case of 'isolated' flight, while the second is due to the contributions associated with formation

flying. The terms $\boldsymbol{C}_{Ff}$ and $\boldsymbol{C}_{Mf}$, are functions of the relative distances between the aircraft. The use of equation (1.31) allows to separate the wake effects in the dynamics model. This can be useful within the design of the control laws, since the wake effects can be seen as disturbances to be rejected by the control system.

For the purposes of the present effort the three degrees-of-freedom, non-linear mathematical model is used:

$$\dot{V} = \frac{(T - D)}{m} - g \sin\gamma \tag{1.32}$$

$$\dot{\gamma} = \frac{g}{V}(n \cos\phi - \cos\gamma) \tag{1.33}$$

$$\dot{\chi} = \frac{g\, n \sin\phi}{V cos\gamma} \tag{1.34}$$

The state variables describing the aircraft dynamics are the airspeed V, the flight path angle γ and the heading angle χ, while T, n and ϕ, are thrust, load factor and bank angle and they are the input variables; D is the aerodynamic drag and m the aircraft mass. According to (1.31), the system can be rewritten introducing changes in lift, induced drag and side force. This leads to:

$$\dot{V} = \frac{T - D}{m} - g \sin\gamma - \frac{\Delta D_i}{m} \tag{1.35}$$

$$\dot{\gamma} = \frac{g}{V}\left[(n + \frac{\Delta L}{mg}) \cos\phi - \cos\gamma \right] \tag{1.36}$$

$$\dot{\chi} = \frac{g(n + \frac{\Delta L}{mg}) \sin\phi}{V \cos\gamma} + \frac{\Delta Y}{mV} \tag{1.37}$$

Thus the general expression of the equation of motion can be rewritten as:

$$\dot{\boldsymbol{X}}_f = \dot{\boldsymbol{X}}_i + \frac{1}{2}\rho V^2 S \left(\boldsymbol{\Delta}_f \cdot \boldsymbol{C}_{Ff}\right) \tag{1.38}$$

where $\boldsymbol{X} = (V, \gamma, \chi)^T$. The subscripts '$f$' and '$i$' indicate 'formation' and 'isolated' flight respectively. Changes in the equations of motion are introduced by the term

$(\boldsymbol{\Delta}_f \cdot \boldsymbol{C}_{Ff})$, where the matrix $\boldsymbol{\Delta}_f \in \mathbb{R}^{(3x3)}$ is defined by:

$$\boldsymbol{\Delta}_f = \begin{pmatrix} 1/m & 0 & 0 \\ 0 & \cos\phi/mV & 0 \\ 0 & \sin\phi/(mV\cos\gamma) & 1/mV \end{pmatrix} \tag{1.39}$$

The coefficients contained in $\boldsymbol{C}_{Ff}$ are functions of the system state vector and the relative position with respect to the other aircraft in the formation.

1.3.2 Formation Kinematics

To maintain the formation geometry, each aircraft must keep its prescribed distance from a reference. Such reference may be the Leader of the formation or a neighborhood aircraft or an imaginary point within the formation. To calculate the relative distance of the $i-th$ aircraft from its reference r, three reference frames are introduced: an inertial, Earth-Fixed frame $\mathbb{F}_{\mathbb{O}}$ and two kinematic frames $\mathbb{F}_{ki}$ and $\mathbb{F}_{kr}$, with the origin on the $i-th$ aircraft and on reference r respectively. The relationship between position derivatives, referred to the inertial frame is given by:

$$\dot{\boldsymbol{P}}_r^O = \dot{\boldsymbol{P}}_i^O + \boldsymbol{V}_{i,r}^O + \boldsymbol{\Omega}_i^O \times \boldsymbol{R}_{i,r}^O \tag{1.40}$$

where $\boldsymbol{P}_{(\cdot)}$ is the position vector, $\boldsymbol{V}_{i,r}$ is the linear velocity of the frame $\mathbb{F}_{ki}$ with respect the frame $\mathbb{F}_{ki}$, $\boldsymbol{\Omega}_i$ is the angular velocity of the frame $\mathbb{F}_{ki}$, and $\boldsymbol{R}_{i,r}$ is the distance vector between the two kinematic frames. By referring to the frame $\mathbb{F}_{ki}$, the previous equation becomes:

$$\boldsymbol{V}_r^i = \boldsymbol{V}_i^i + \boldsymbol{V}_{i,r}^i + \boldsymbol{\Omega}_i^i \times \boldsymbol{R}_{i,r}^i \tag{1.41}$$

where $\boldsymbol{V}_{i,r}^{i} = [\dot{dx},\, \dot{dy},\, \dot{dz}]^T$, $\boldsymbol{\Omega}_i^i = [0,\, \dot{\gamma}_i,\, \dot{\chi}_i]^T$, and $\boldsymbol{R}_{i,r}^{i} = [dx,\, dy,\, dz]^T$ while the vector $\boldsymbol{V}_r^i$ is given by:

$$\boldsymbol{V}_r^i = \begin{bmatrix} V_r \cos\gamma_e \cos\gamma_e \\ V_r \cos\gamma_e \sin\chi_e \\ -V_r \sin\gamma_e \end{bmatrix} \tag{1.42}$$

where $\gamma_e = \gamma_r - \gamma_i$ and $\chi_e = \chi_r - \chi_i$. Thus the complete dynamic of the distance between the $i-th$ aircraft and its reference is given by:

$$\begin{bmatrix} \dot{dx} \\ \dot{dy} \\ \dot{dz} \end{bmatrix} = \begin{bmatrix} V_r \cos\gamma_e \cos\gamma_e \\ V_r \cos\gamma_e \sin\chi_e \\ -V_r \sin\gamma_e \end{bmatrix} - \begin{bmatrix} V_i \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \dot{\gamma}_i dy - \dot{\chi}_i dz \\ -\dot{\chi}_i dx \\ \dot{\gamma}_i dx \end{bmatrix} \tag{1.43}$$

The previous method for relative distances calculation is typically used when aircraft within the formation exchange trajectory information, particularly airspeed, flight path angle, and heading angle.

If the trajectory information are not available, and the if aircraft knows the absolute position (e.g. by GPS) only, the relative distances between aircraft could be computed first referred to the inertial frame and then rotated to the kinematic frame of each aircraft. The distance between the $i-th$ aircraft from its reference r, referred to the inertial frame is defined as:

$$\boldsymbol{d}_i^O = \boldsymbol{P}_r - \boldsymbol{P}_i \tag{1.44}$$

where $\boldsymbol{P}_i$ and $\boldsymbol{P}_r$ are the position of the $i-th$ aircraft and its reference in the Earth-fixed frame F_O. In formation flying it is necessary to rotate the distance between aircraft and their reference in a local frame which has its origin in the center of mass of each aircraft. The transformation of the inertial, Earth-fixed frame F_O to the

Kinematic frame $\mathbb{F}_{ki}$ can be obtained introducing the following rotation matrix: 1

$$T_{Ok_i} = \begin{pmatrix} \cos\chi_i \cos\gamma_i & \sin\phi_i \sin\gamma_i \cos\chi_i & \cos\phi_i \sin\gamma_i \cos\chi_i \\ & -\sin\chi_i \cos\phi_i & +\sin\phi_i \sin\chi_i \\ \sin\chi_i \cos\gamma_i & \sin\gamma_i \sin\phi_i \sin\chi_i & \sin\gamma_i \cos\phi_i \sin\chi_i \\ & +\cos\chi_i \cos\phi_i & -\sin\phi_i \cos\chi_i \\ -\sin\gamma_i & \cos\gamma_i \sin\phi_i & \cos\gamma_i \cos\phi_i \end{pmatrix} \tag{1.45}$$

Thus distance of the $i-th$ aircraft from its reference in the Kinematic frame $\mathbb{F}_{ki}$ is given by:

$$\boldsymbol{d}_i^k = \mathbb{T}_{O_{ki}}^T \boldsymbol{d}_i^O \tag{1.46}$$

Chapter 2

Formation Flight Control

2.1 Formation Flight Control

As stated in the introduction, one of the main current approaches to the problem of control and coordination of multi-vehicle formations is represented by the Leader/Wingman structure. Within such approach, one of the aircraft (Leader) is chosen to direct the formation, following a prescribed path (or a Virtual Leader) and all the other aircraft (Wingmen) are expected to maintain a fixed relative distance to a reference.

Let us now consider two variations of Leader/Follower structure: the first where the reference is a neighborhood aircraft and the second where the reference is the Leader. In the first case, the rear Wingmen exhibit a poorer transient response with respect the front ones due to error propagation while, in case of each Follower takes the trajectory reference from the Leader of formation, such configuration may be critical because each Follower, that is directly connected to the Leader, has no information about its distance from the other Followers, and therefore it would not be capable of

avoiding collision. Thus, Leader/Follower structure may not be the best strategy to perform complicated maneuvers, especially in multi-aircraft formation flight. In fact, each aircraft in the formation must cooperate in order to maximize the possibilities for a formation to obtain and retain its structure. The aircraft in the formation must work together to achieve a common goal.

To overcome the limitations of Leader/Wingman structure, a different strategy is proposed, based on the behavior of migratory birds [Anderson and Robbins, 1998]. The aircraft in a formation are not longer referring to each other, but they are required to keep a specified distance from an imaginary point called Formation Geometry Center (FGC). The FGC position depends on the relative distances between the aircraft in the formation itself; this allows each aircraft to have the capability of sensing other vehicles movement from the nominal position in the formation. In the presence of disturbances, for instance, if one of the aircraft loses its position, the other senses the change, and depart momentarily from the prescribed trajectory, manoeuvreing all-together in order to reconstitute the formation geometry. Once the geometry has been reached again, all aircraft continue to follow the prescribed trajectory.

2.1.1 Natural Behavior of Migratory Birds

For long distance migrational flight, birds of several species tend to move in a formation, flying close to each other maintaining a defined geometrical shape. One of the most interesting considerations is the fact that during a migrational flight if one or more elements of the group loses its position in the formation, the others leave the migration trajectory and 'wait' for the lost ones until the formation shape is reconstituted.

There are two main reasons that birds fly in a formation; the first reason is related to aerodynamic considerations while the other is due to the 'social behavior' of some species (see reference [Tyne and Berger, 1976]). In terms of aerodynamics effects, birds take advantage of the induced velocity produced by the wing tip vortex phenomenon: the inner wing of each bird in a V formation, for example, gains an increase in lift and then a reduction in induced drag from the upward rising side of the vortex left by the outer wing of the bird ahead. For this reason, it is important that each bird in the group keep its position; losing position by one or more birds in a migrational flight means a non-negligible of aerodynamics efficiency loss. On the other hand, birds like geese and swans form 'family' groups: offsprings remain with parents one or two seasons after the birth and fly together with them during the migration. The elements in these formations know each other and try to remain together independently from aerodynamics advantages while other birds like storks live in non-familiar group, but fly together with others just in order to improve the efficiency of the migration (especially during food searching or mating season). From these motivations it's possible to assume that the preceding considerations to be reasonable. A two-aircraft formation is considered in the present work. Each aircraft will be represented using the modified three-degrees of freedom point-mass model described in the previous section.

2.1.2 Formation Geometry Center

As described in the previous section, aircraft do not refer to each other and, in order to maintain formation geometry an imaginary point called Formation Geometry Center (FGC) is introduced and each aircraft must keep a prescribed distance from this point. The FGC dynamics for an n_a-aircraft formation can be represented by the following

differential equations:

$$\dot{x}_{FGC} = \sum_{i=1}^{n_a} \frac{V_i \cos\gamma_i \cos\chi_i}{n_a} \tag{2.1}$$

$$\dot{y}_{FGC} = \sum_{i=1}^{n_a} \frac{V_i \cos\gamma_i \sin\chi_i}{n_a} \tag{2.2}$$

$$\dot{z}_{FGC} = \sum_{i=1}^{n_a} \frac{-V_i \sin\gamma_i}{n_a} \tag{2.3}$$

By integrating the preceding equations, the position of the FGC in the Earth-fixed frame ($\boldsymbol{P}_{FGC}$) is found and the distance between the FGC and the $i-th$ aircraft, referred to the frame $\mathbb{F}_{\mathbb{O}}$, is defined as:

$$\boldsymbol{d}_i = \boldsymbol{P}_{FGC} - \boldsymbol{P}_i \tag{2.4}$$

where $\boldsymbol{P}_{FGC}$ and $\boldsymbol{P}_i$ are the position of the FGC and the $i-th$ aircraft respectively. The transformation of the distance vector from the inertial, Earth-fixed frame F_O to the Kinematic frame F_{k_i} is modelled by equation (1.46).

2.1.3 Inner-Loop Synthesis

Prior to the design of any formation controller, each aircraft of the formation must feature adequate tracking of a commanded trajectory capabilities through an inner loop controller. The inner-loop controller is based on the model for an individual aircraft in undisturbed air described in the previous section. The following three lag filters are introduced to model the fact that the aircraft cannot change bank angle (ϕ), load factor (n) or thrust (T) instantaneously.

$$\dot{T} = (T_c - T)/\tau_t \tag{2.5}$$

$$\dot{\phi} = (\phi_c - \phi)/\tau_b \tag{2.6}$$

$$\dot{n} = (n_c - n)/\tau_n \tag{2.7}$$

T_c, ϕ_c and n_c are the commanded values for input variables. The time constants τ_t, τ_b, and τ_n will be aircraft specific. To implement the inner-loop controller, a dynamic inversion law is used, [Anderson and Robbins, 1998]. The desired trajectory vector is written in terms of a commanded airspeed (V_c), flight path angle (γ_c), and heading angle (χ_c). It is assumed that the desired trajectory will be governed by the following linear time-invariant equations:

$$\dot{V} = \omega_V(V_c - V) \tag{2.8}$$

$$\dot{\gamma} = \omega_\gamma(\gamma_c - \gamma) \tag{2.9}$$

$$\dot{\chi} = \omega_\chi(\chi_c - \chi) \tag{2.10}$$

where ω_V, ω_γ and ω_χ are the selected bandwidths of the controller and they dictate how quickly the actual aircraft trajectory changes with respect to the new desired value. To derivate the dynamic inversion control law, the commanded rates (2.8)-(2.10) are set equal to the associated actual rates (1.35)-(1.37), and solving for thrust, bank angle, and load factor. The computed thrust, bank angle, and load factor become the commanded values needed for (2.5)-(2.7).

The airspeed command is obtained by setting equation (2.8) equal to (1.35) and solving for thrust. This yields:

$$T_c = D + \omega_v(V_c - V)m + mg\sin\gamma \tag{2.11}$$

Substituting equation (2.9) in (1.36),

$$n_c\cos\phi_c = \frac{V}{g}\left[k_\gamma\left(\gamma_c - \gamma\right) + \cos\gamma\right] = c_1 \tag{2.12}$$

and (2.10) into (1.37) leads to:

$$n_c \sin \phi_c = \frac{V}{g} k_\chi \left(\chi_c - \chi\right) \cos \gamma = c_2 \tag{2.13}$$

Finally, equations (2.12) and (2.13) can be combined to yield the required load factor and bank angle commands:

$$n_c = \sqrt{{c_1}^2 + {c_2}^2} \tag{2.14}$$

and

$$\phi_c = tan^{-1}\left(\frac{c_2}{c_1}\right) \tag{2.15}$$

Now the aircraft/autopilot model is obtained using the equations derived above, the model for a n-aircraft formation is obtained by introducing the force coefficient variations and the relative distances dynamics:

$$\begin{aligned}
\dot{\boldsymbol{x}}_1 &= f_1(\boldsymbol{x}_1, \boldsymbol{u}_1) + \tfrac{1}{2}\rho V_1^2 S_1 \left(\boldsymbol{\Delta}_{f_1} \cdot \boldsymbol{C}_{F_1 f}\right) \\
&\cdots \\
\dot{\boldsymbol{x}}_{n_a} &= f_2(\boldsymbol{x}_{n_a}, \boldsymbol{u}_{n_a}) + \tfrac{1}{2}\rho V_{n_a}^2 S_{n_a} \left(\boldsymbol{\Delta}_{f_{n_a}} \cdot \boldsymbol{C}_{F_{n_a} f}\right) \\
\boldsymbol{d}_1 &= g_1(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n_a}) \\
&\cdots \\
\boldsymbol{d}_{n_a} &= g_{n_a}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n_a})
\end{aligned} \tag{2.16}$$

where $f_{(\cdot)}$ contains the aircraft/autopilot model and $\boldsymbol{d}_i$ is the distance between the i-th aircraft and the FGC. This is the complete nonlinear point-mass model that is used in simulation.

2.1.4 Formation Controller Design

A Formation Controller (FC) is needed to reproduce the natural behavior of migrational birds. For the purpose of FC design, linearization around a selected flight

condition of the previous nonlinear model yields the linearized set of equation described by the following state space model:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} A_1 & \dots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{0} & \dots & A_{n_a} \end{bmatrix} x + \begin{bmatrix} B_1 \\ \vdots \\ B_{n_a} \end{bmatrix} u \\ y &= \begin{bmatrix} C_1 & \dots & C_{n_a} \end{bmatrix}^T x \end{aligned} \tag{2.17}$$

The Formation Controller (FC) is given by two separate control systems: a trajectory controller (K_T) and a position controller (K_P). The main objective of the trajectory controller is to provide tracking of a prescribed path in terms of desired velocity, flight path and hading angle: r_T=(V_d, γ_d, χ_d), while the position controller is designed to maintain inter-aircraft distances in order to give the formation a desired geometric shape. Trajectory and position controllers receive path error information, that is the difference between current and desired path, and FGC-distance error, that is the difference between current and desired distance from FGC, respectively:

$$\begin{aligned} e_T &= r_T - y_T \\ e_P &= r_P - y_P \end{aligned} \tag{2.18}$$

and they generate trajectory and position commanded vectors u_T and u_P defined as:

$$\begin{aligned} u_T &= K_T e_T \\ u_P &= K_P e_P \end{aligned} \tag{2.19}$$

For system (2.17) two LQ-Servo controllers were developed providing gain matrices K_T and K_P. Such matrices were obtained through the minimization of the quadratic cost indexes:

$$\begin{aligned} J_{(\cdot)} = \int_{t_0}^{\infty} \big[& u' \, R_{(\cdot)} \, u + e'_{(\cdot)} \, Q_{e(\cdot)} \, e_{(\cdot)} + \\ & + x'_{r(\cdot)} Q_{r(\cdot)} x_{r(\cdot)} \big] dt \end{aligned} \tag{2.20}$$

where Q_e, Q_r and R are positive definite matrices. They regards the minimization of the desired output vector, residual state vector and input vector respectively. The following assumptions are introduced:

$$\bar{C} = I - L\,C \tag{2.21}$$

$$L = C'\,(C\,C')^{-1} \tag{2.22}$$

These relationship allows to rewrite equation (2.20) as:

$$J_{(\cdot)} = \int_{t_0}^{\infty} \left[u'\,R_{(\cdot)}\,u + (x - \tilde{x})'\,Q_{(\cdot)}\,(x - \tilde{x})\,\right] dt \tag{2.23}$$

where:

$$Q_{(\cdot)} = \bar{C}'Q_{r(\cdot)}\bar{C} + C'Q_{e(\cdot)}C \tag{2.24}$$

$$\tilde{x} = L \begin{bmatrix} r_T \\ r_P \end{bmatrix} \tag{2.25}$$

The resulting commanded vector u_c is a convex combination of the two previous control laws:

$$u_c = \eta u_T + (1 - \eta) u_P \tag{2.26}$$

where $\eta \in [0, 1]$, may be a constant or a function. The prescribed path can be commanded by one of the aircraft of the formation or by a ground station o by a Virtual Leader [G.Mancino *et al.*, 1999].

Assume η to be an exponential function of position error e_P:

$$\eta = \theta_1 \exp\left(-\frac{|e_p|}{\theta_2}\right) \tag{2.27}$$

During a perturbed flight , there may be one or more aircraft with a position error with respect to the nominal configuration. This cause the FGC move to a new

position and then all the aircraft within the formation have a position error e_P. The command vector u_c, depends both on position error e_P and trajectory error e_T. and so the aircraft in the formation may not be able to follow the prescribed path exactly until the geometry is re-established. When the formation has the desired shape, the position error e_P is zero and η=1 for each aircraft . This causes the commanded vector u_c be given only by the contribution of the trajectory controller K_T and all the aircraft will follow the desired trajectory r_T, allows to achieve a cooperative behavior. By varying the function η each aircraft could have its own behavior. The stability of the control law expressed by Equation (2.26) is provided by the following proposition.

(Positive Real Lemma) Given the LMI problem:

$$P > 0, \quad \begin{bmatrix} A^T P + PA + Q & PB \\ B^T P & R \end{bmatrix} \leq 0 \tag{2.28}$$

where $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{p \times n}$ and $P = P^T \in \mathbb{R}^{n \times n}$. If $R = R^T$ and $Q = Q^T$ are positive definite, the LMI problem (2.28) is equivalent to the following quadratic matrix inequality:

$$A^T P + PA + Q - PBR^{-1}B^T P \leq 0 \tag{2.29}$$

Given a system, whose dynamics is modeled by the standard state space model (A, B, C, D) and given the matrices Q_1, Q_2, and R positive definite. If exist two matrices P_1 and P_2 positive definite so that the two optimal controllers $K_1 = -R^{-1}B^T P_1$ and $K_2 = -R^{-1}B^T P_2$ stabilize the system, then the convex combination of the two controllers, $K = \alpha K_1 + \beta K_2$, stabilizes the system as well.

Proof: If controllers K_1 and K_2 stabilizes the system then, for the previous

proposition:

$$\begin{bmatrix} A^T P_1 + P_1 A + Q_1 & P_1 B \\ B^T P_1 & R \end{bmatrix} \leq 0 \tag{2.30}$$

and:

$$\begin{bmatrix} A^T P_2 + P_2 A + Q_2 & P_2 B \\ B^T P_2 & R \end{bmatrix} \leq 0 \tag{2.31}$$

Let now consider the convex combination of (2.30) and (2.31):

$$\begin{bmatrix} \alpha\,(A^T P_1 + P_2 A + Q_1) + \beta(A^T P_2 + P_2 A + Q_2) & (\alpha P_1 + \beta P_2) B \\ B^T(\alpha P_1 + \beta P_2) & R \end{bmatrix} \leq 0 \tag{2.32}$$

According to Proposition 1:

$$\alpha\,(A^T P_1 + P_2 A + Q_1) + \beta(A^T P_2 + P_2 A + Q_2) \leq (\alpha P_1 + \beta P_2) B R^{-1} B^T (\alpha P_1 + \beta P_2) \tag{2.33}$$

and thus:

$$A^T P + P A \leq P B R^{-1} B^T P - (\alpha Q_1 + \beta Q_2) \tag{2.34}$$

where $P = \alpha P_1 + \beta P_2$. Let now choose the following Lyapunov function:

$$V(x) = x^T \widetilde{P} x \tag{2.35}$$

where $\widetilde{P}$ is positive definite. The time derivative is:

$$\dot{V}(x) = \dot{x}^T \widetilde{P} x + x^T \widetilde{P} \dot{x} \tag{2.36}$$

and it must be negative defined, to guarantee stability:

$$\dot{V}(x) = x^T [A^T - B R^{-1} B^T (\alpha P_1 + \beta P_2)] \widetilde{P} x + x^T \widetilde{P} [A - B R^{-1} B^T (\alpha P_1 + \beta P_2)] x \tag{2.37}$$

By choosing $\widetilde{P} = \alpha P_1 + \beta P_2$, and introducing the inequality in equation 2.34 leads to:

$$\begin{aligned}\dot{V}(x) &= x^T[A^T(\alpha P_1 + \beta P_2) + (\alpha P_1 + \beta P_2)A]x - 2x^T[(\alpha P_1 + \beta P_2)BR^{-1}B^T(\alpha P_1 + \beta P_2)]x \\ &\leq x^T[-\alpha Q_1 - \beta Q_2 - (\alpha P_1 + \beta P_2)BR^{-1}B^T(\alpha P_1 + \beta P_2)]x\end{aligned} \tag{2.38}$$

All the terms are negative, then the convex combination of the two controllers K_1 and K_2 stabilize the system.

□

2.2 Simulation Results

To validate the overall control strategy, simulations were performed for close formations consisting of two unmanned aircraft as shown in Figure 2.1. The aircraft model includes aircraft/autopilot model, the Distributed Horse-Shoe Vortex code and FGC relative distance dynamics. The Formation Control scheme is shown in Figure 2.2 and is simulated in MATLAB environment. The relative nominal distances between aircraft and FGC are selected to be 5 m along the x and y-axes. The altitude is the same for both aircraft at 300 m.

The first simulation is a path following task. Each aircraft receives the commanded trajectory from the VL. Starting from a steady-level flight condition at 20 m/sec, each aircraft is commanded to perform a heading change of -30 deg. Trajectory tracking is achieved and the relative distances between the aircraft are maintained at nominal values. Figure 2.3 shows aircraft response to the heading command in case of Leader/Wingman structure and FCG-based structure. The figure shows how the error propagation is reduced. Figure 2.4 and 2.5 show the relative distance between

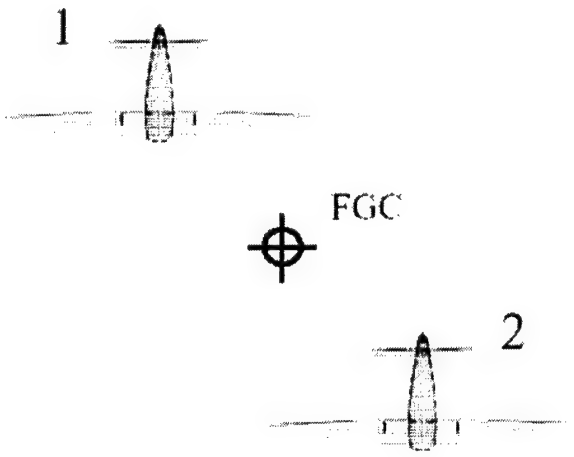


Figure 2.1: Formation Geometry with FGC

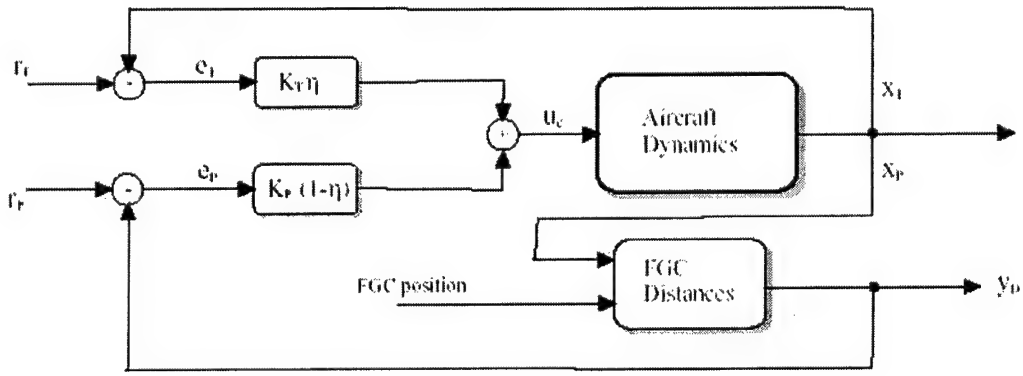


Figure 2.2: Formation Controller

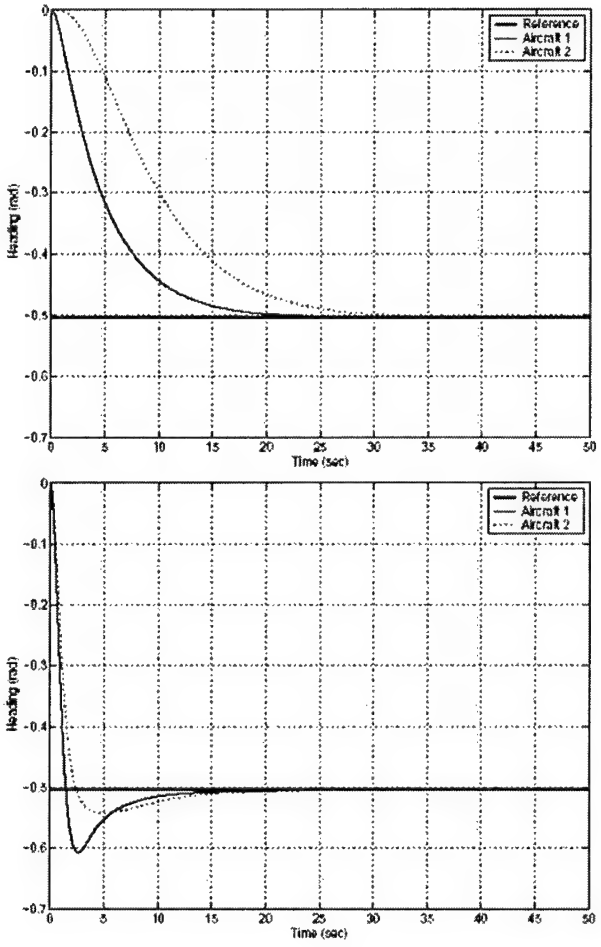


Figure 2.3: Aircraft Responses to a -30 deg Heading Change in a L-W Structure

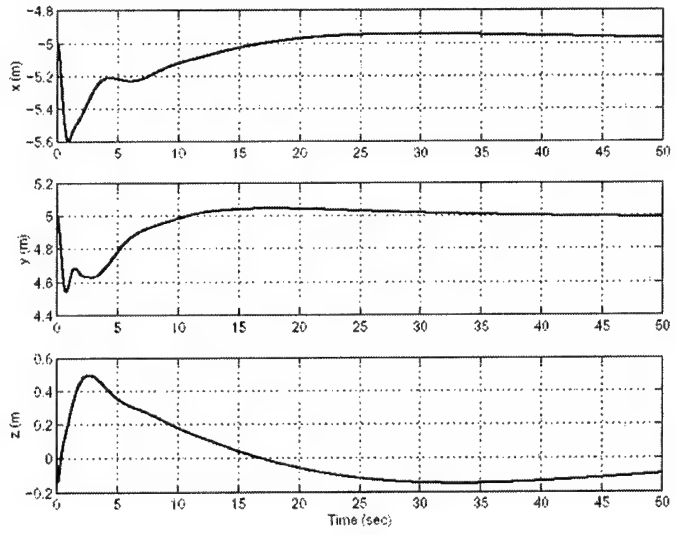


Figure 2.4: Aircraft 1, FGC Distances during Heading Change

aircraft and the FGC during the heading change manoeuvre (FGC-based structure).

The second simulation is a formation geometry variation. Each aircraft is commanded to increase its relative distance from FGC from 5 to 7 m along x and y-axis. Figure 2.6 and 2.7 show the relative distance between aircraft and the FGC while Figure 2.8 shows the airspeed time history during such manoeuvre.

The last simulation shows how the FC could reproduce the behavior of migrational birds. The initial condition is a steady-level flight. Each aircraft has an initial airspeed of 20 m/sec that is the commanded trajectory. A speed perturbation is introduced to aircraft 2 and it slows down with respect to aircraft 1. Figure 2.9 shows speed profiles for the two aircraft. It is clear how aircraft 1 stops following the prescribed velocity and 'waits' for aircraft 2 in order to maintain the nominal distances. Figure

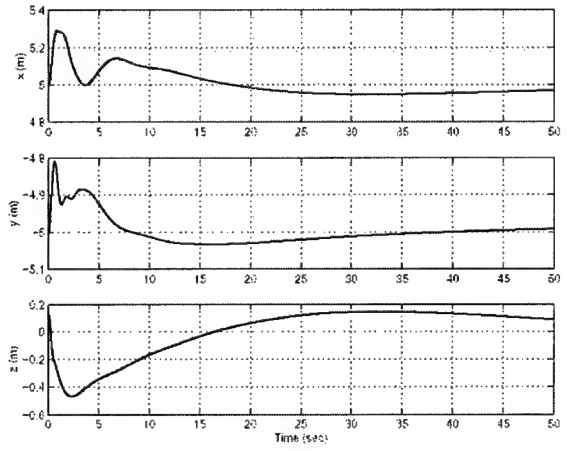


Figure 2.5: Aircraft 2, FGC Distances during Heading Change

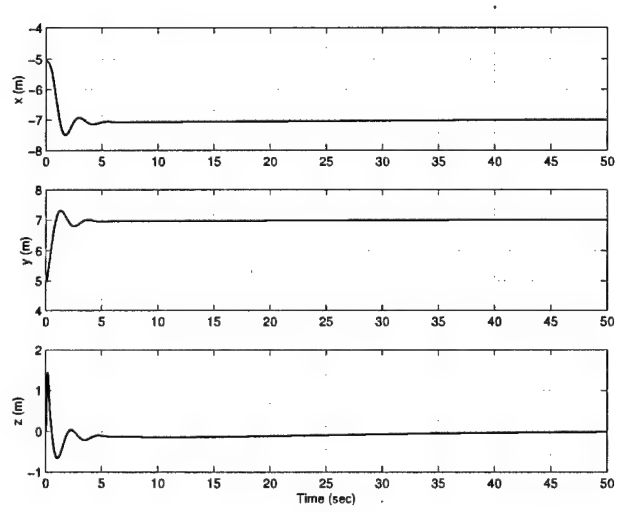


Figure 2.6: Aircraft 1-FGC relative Distance during Geometry Variation

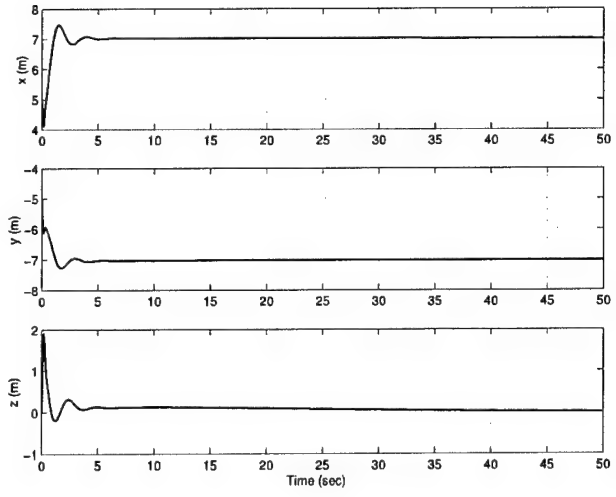


Figure 2.7: Aircraft 2-FGC relative Distance during Geometry Variation

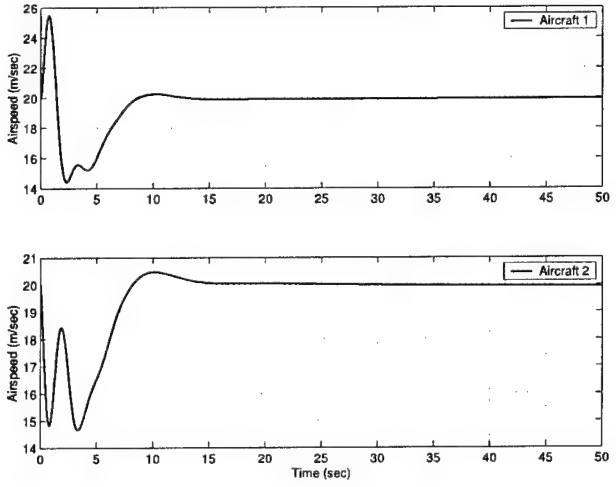


Figure 2.8: Airspeed Time History during Geometry Variation

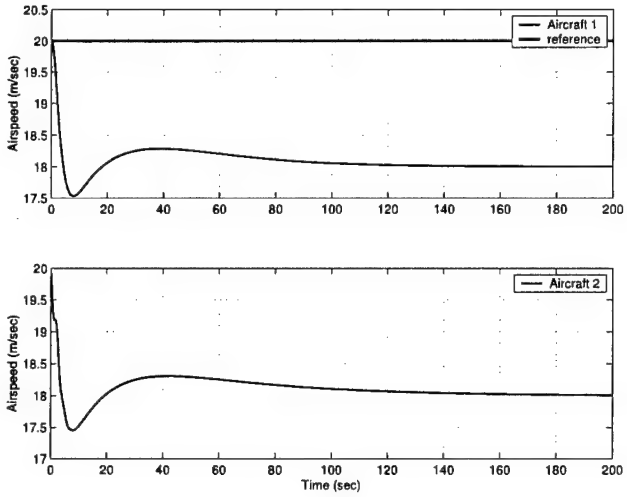


Figure 2.9: Airspeed Time History during perturbed Flight

2.10 shows the $x-$axis relative distance between aircraft during perturbed flight.

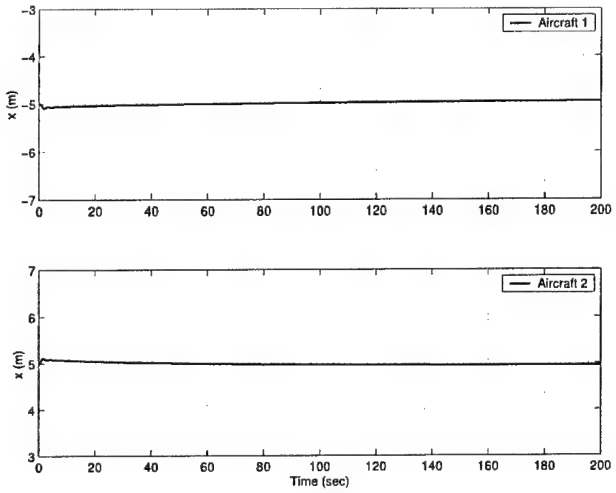


Figure 2.10: X-Distance from FGC during perturbed Flight

Chapter 3

Reconfiguration in the presence of Communications Failures

Coordination and management within a formation of multiple unmanned air vehicles (UAV) is a critical issue for utilizing their full potential in operational situations. While military and civilian mission scenarios would benefit from the addition of autonomous individual UAV, compared to non-autonomous or manned aircraft, it is clear that formation flight, with an effective coordination strategy, could lead to superior performance and resources utilization.

Coordinated control of multiple vehicles has been widely studied in the past. Kangl and co-workers addressed the problem of formation coordination and reconfiguration of multiple micro satellites, [Mesbahi, M. and Hadaegh, F. Y., 1999] and [Meshabi, M. and Hadaegh, F. Y., 2001] approached the Leader/Wingmen structure for multiple spacecraft with graph theory and linear matrix inequalities (LMI) techniques, while [Godbole, D. N. *et al.*, 2000] and others presented a communication protocol for the operation of an automated highway system in the presence of failures.

Flight coordination of multiple UAV was recently addressed by [F. Giulietti *et al.*, 2000], McLain and others [McLain, T. W. *et al.*, 2000] presented results on the decentralized trajectory planning for coordinated rendezvous of multiple air vehicles. In this section we investigate possible formation structures to be used in an operational environment, with emphasis on optimization with respect to data information transfer among the elements of the formation itself.

3.1 Formation Flight Management

The main issues involved in control and management of an aircraft formation are trajectory tracking and inter-aircraft relative distance regulation. In the case of decentralized management, each vehicle in the formation needs to exchange data, such as position and trajectory information. Many different configurations could be found for the communication flow within a formation, and not all of these configurations may provide good performance. Thus, the first step is to find an "optimal" one, among different configurations for information exchange.

Optimization of available communication channels based on a cost function figure of merit can be achieved using a variety of methods; here the problem is set up using graph-programming techniques. The formation is viewed as an oriented graph: the nodes represent the aircraft, while the physical communication channels between vehicles create the arcs. The graph is oriented because, in the most general case, channels are not bi-directional; this is not a limitation, since two nodes can be connected by two opposite direction arcs to model a bi-directional channel. The graph must also be connected because, if two sub-graphs exist without any arc connecting them, the aircraft in the two groups cannot behave as a single formation, but rather

act as two separate formations.

In the optimization process, each arc must be given a weight. The optimization will minimize the total cost of the information paths throughout the formation using the arcs weights to evaluate the cost of a connection. The weights for the arcs can be selected by taking into account general mission requirements related to the formation flight such as:

- Closed-loop performance: a measure of the capability of the formation control system

to maintain the prescribed path and the nominal inter-aircraft distances, may be used to set the weight of the arc.

- Formation safety: communications between neighborhood aircraft will have a lower

cost. Using distance references with adjacent aircraft might limit the risk of aircraft conflicts compared to using a common reference for all aircraft.

- Type of communication channel: in case a non-radio-based (or a non-omnidirectional)

communication channel is used, the geometry of the formation might influence the possibility of exchanging data between two airplanes that are not closely spaced or that are hidden by other vehicles.

In general, since the position error propagates and increases throughout the formation, the optimization algorithm should include the minimization of the minimum error propagation path. Once an optimal solution for the communication flow is found, the next step in the formation management design consists in giving the structure adequate robustness to communication failures. A failure in the communication occurs when one or more aircraft of the formation loses the information exchange capability

(send or receive data). Obviously the loss of the aircraft itself is a communication failure: both receiving and sending information capabilities are lost. After a failure, there may be one or more connections lost and then a new channel configuration must be found. The resulting configuration will not be optimal, because of the loss of one or more channels, but it will be the optimal solution according to the new set of nodes and arcs. The algorithm for the re-optimization of the inter-aircraft connection is triggered by the fault detection, and it is decentralized for faster reconfiguration time, and information other than that strictly needed for the formation-keeping control system must be exchanged on the data channels. The reconfiguration process must be the same for all aircraft; that is, the local copy of the graph describing the formation communications must be identical in all aircraft at all times.

There are cases in which, after a reconfiguration of the inter-aircraft connections, a geometrical reconfiguration of the formation is needed as well. Such cases are: aircraft loss, where the geometrical reconfiguration is related to aerodynamic efficiency effects, the data receiving capability loss for a 'leaf' aircraft of the graph or the data sending capability loss by the root of the graph. In the latter cases, the geometrical reconfiguration is needed to keep the information flow (under the hypothesis that information flows from the front aircraft towards the rear aircraft). The geometrical reconfiguration of the formation is described by a set of heuristic rules that are implemented through several schemes, called reconfiguration maps (RM), presented later.

Figure 3.1 shows the complete control and management architecture that will be used as reference in the rest of the paper.Where:

BC=Broadcast Channel

RM = Reconfiguration Manager

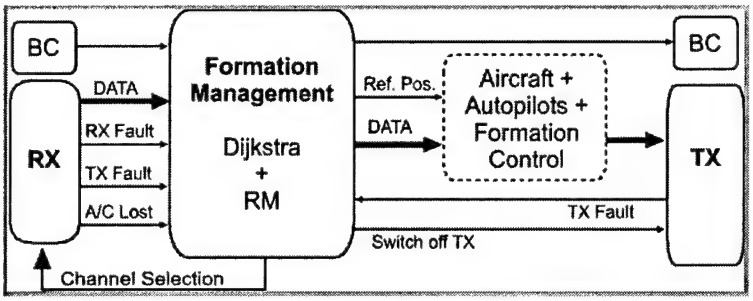


Figure 3.1: Reference Management and Formation Control Architecture

RX = Receiver

TX = Transmitter

The gray block in the diagram represents the aircraft dynamics, the inner loop autopilots, and the formation flight control system. This single block is actually a two-loop system, with three inner loop autopilots for tracking of commanded speed, flight path and heading angles, and an outer loop (formation control) responsible for trajectory following and inter-aircraft distance control. For the purpose of the present work, a three-degree of freedom (3-DOF) point mass model was used. Using standard aerospace notation:

$$\dot{V} = \frac{(T - D)}{m} - g \sin\gamma \tag{3.1}$$

$$\dot{\gamma} = \frac{g}{V}(n \cos\phi - \cos\gamma) \tag{3.2}$$

$$\dot{\chi} = \frac{g\, n \sin\phi}{V cos\gamma} \tag{3.3}$$

Assuming perfect modeling, aircraft dynamics can be feedback linearized with the following control laws:

$$T = k_v(V_d - V)m + mg\sin\gamma + D \tag{3.4}$$

$$n\cos\phi = \frac{V}{g}\left[k_\gamma\left(\gamma_d - \gamma\right) + \cos\gamma\right] = c_1 \tag{3.5}$$

$$n\sin\phi = \frac{V}{g}k_\chi\left(\chi_d - \chi\right)\cos\gamma = c_2 \tag{3.6}$$

the resulting linear system becomes:

$$\begin{cases} \dot{V} = K_V\left(V_d - V\right) \\ \dot{\gamma} = K_\gamma\left(\gamma_d - \gamma\right) \\ \dot{\chi} = K_\chi\left(\chi_d - \chi\right) \end{cases} \tag{3.7}$$

To maintain the desired formation geometry, each aircraft must keep its relative position within the formation. The formation management is derived using Dijkstra's algorithm3, and reconfiguration maps give the reference position, according to the results of the optimization procedure. In the remainder, a formation of 6 aircraft will be used as example (see Figure 3.2).

3.2 Optimal Communications

As outlined in the previous section, an aircraft formation is described using graph theory. The presence of an outgoing arc in a node in the graph implies the capability of transmitting information, while an incoming arc is related to the capability of receiving information. Two virtual devices model such capabilities: a transmitter (TX) and a receiver (RX). These devices are "virtual" in the sense that failure implies the loss of the device capability, irrespectively of what subcomponent has actually failed (antennas, CPU, transmission bus, etc.). In the presence of TX and/or

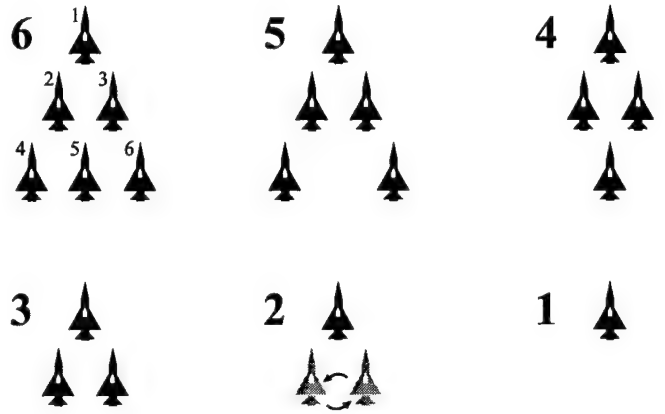


Figure 3.2: Allowed Formation Geometries

RX failure all the outgoing and/or incoming arcs are lost, while in case of aircraft loss, all incoming and outgoing arcs from the node representing the lost aircraft are neglected. It is important to notice that considering TX and RX separately, allows the extension of the present technique to non-radio-based communication devices.

3.2.1 The Virtual Leader

During the mission, the reference trajectory could be provided in two ways. It may be stored in one (or more) on-board computers, or it may be provided remotely from a ground station as well as from a manned aircraft. In both cases, not all the aircraft may be required to know the reference trajectory. However, according to the properties of the Leader/Wingman structure5, at least one aircraft in the formation knows the reference trajectory. The reference trajectory can be seen as the effective leader of the formation, by introducing an imaginary point moving in the space, tracking the path prescribed for the formation and rigidly followed by all the aircraft

of the formation. Such imaginary point is called Virtual Leader (VL) and can be formalized through the following definitions:

Definition 1: The nodes representing the aircraft of the formation belong to the set :$L = \{v_1, v_2, ..., v_n\}$,n = number of aircraft

Definition 2: The nodes representing the aircraft of the formation knowing the reference trajectory, belong to the subset of $V : L = \{l_1, l_2, ..., l_p\}$,$p \leq n$.

Definition 3: The graph $F = (V, E)$, where E is a finite set of arcs, is a formation graph (FG) if and only if: $\forall v \in (V/L) \exists l \in L$, Such that one of the following conditions hold:

Such that one of the following conditions hold:

1. $(l, v) \in E$

2. $\exists C = \{v1, v2, ..., vk\}, k \geq 1, C \subseteq (V/L) \in E \wedge (\forall i = 1, ..., k-1, (vi, vi+1) \in E)$.

Definition 4: Given a node n, the graph $F\prime = (V\prime, E\prime)$, where $V' = V \smile v', E' = E \smile \widetilde{E}$, and $\widetilde{E}$ is a finite set of arcs, is an extended formation graph (EFG) if and only if:

$F = (V, E)$ is a formation graph (FG)

1. $\widetilde{E}$ contains every arc outgoing from the node n to the nodes belonging to the set L

2. The nodev' is called Virtual Leader (VL).

Definition 5: An EFG is defined feasible if and only if it is connected and the nodes have at most one incoming arc.

Once the VL has been defined, the reference trajectory can be inserted in the optimal communication configuration procedure.

3.2.2 Graph Theory Approach

The send-receive nature of communications leads to an oriented graph; the direction of the arcs indicates the direction of data flow and the available communication channels can form cycles8; with a broadcast communication scheme, for instance, all possible arcs exist among nodes. Furthermore, the arcs capacity is unlimited; there is no evident physical meaning of a communication channel with limited "capacity." The arcs weight can be set, without loss of generality, to values greater than zero. The following propositions are useful in the optimization procedure.

: If an EFG contains a cycle, the cycle does not contain VL.

: Suppose the VL is included in the cycle. From the definition of cycle node, the VL should have an outgoing arc and an incoming arc. This is in contrast with the condition 2 of Def. 4. Thus the VL is not included in the cycle. ■

: A feasible EFG contains no cycles.

: Suppose that a cycle exists. VL is not included in the cycle (Prop. 1) but it must be connected to the cycle (Def. 5). The VL may not be connected directly to the cycle because both cycle and VL admit only outgoing arcs. If nodes were added to create a path between the cycle and the VL, they also could have at least one incoming arc (Def. 5). Then there is no additional node that could receive the outgoing arc from the VL. Thus EFG does not contain any cycle, and the VL will be the root of the solution tree. ■

Under these assumptions, the problem can be configured as a shortest path problem (SPP), and Dijkstra's algorithm is used, as outlined in Figure3.3.

Each node in the graph represents a position in the formation, not necessarily an aircraft. Node i has a potential $d(i)$ and a preceding node $p(i)$. The potential is a temporary value used by the algorithm and is initialized to + infinity, except for the

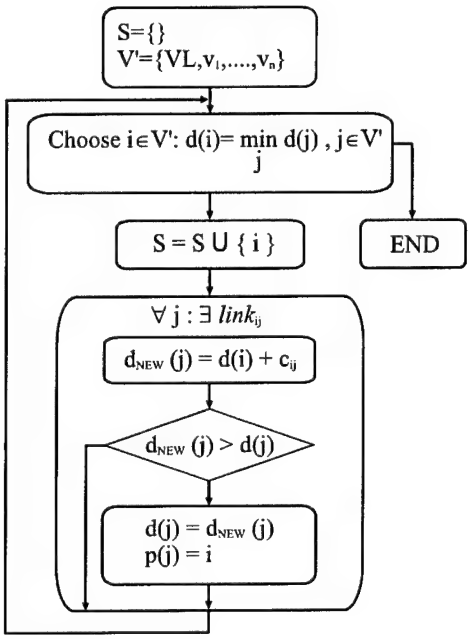


Figure 3.3: Dijkstra's Algorithm Flow Chart

VL that has zero potential. For all the nodes the initial value for the preceding node is the VL. The S set is called the definitive-label set and V' is called the temporary-label set. Each arc, going from node i to node j is assigned the weight Cij.

The algorithm chooses, in the temporary-label set $V\prime$, the node that has the minimum potential and moves it to the S set until the V' set is empty. Then, for all outgoing arcs, the actual node computes a tentative new potential for the arrival node. If this new potential is lower than the previous potential, it updates that node potential and sets its preceding node to itself. The algorithm runs until all nodes have been assigned a potential. This solution provides the minimum non-cyclic path connecting all nodes. No information is available at this point about possible sub-optimal arc sets.

3.3 Communication Failures

The communication system is modeled with a TX and a RX device on each aircraft. A TX or a RX can become faulty at any time, and a fast reconfiguration of the communication channels is needed. The activation of a new communication channel between two nodes in place of the broken one is determined by the presence of a working TX and a working RX on the aircraft occupying the nodes, therefore it depends on which communication terminal breaks down, and on the history of past communication failures.

Each RX device must be able to reliably detect when a communication channel is lost. This means when the corresponding TX has become faulty, the reference aircraft has left the formation, or its own RX is not functioning. While the RX device is deciding whether the channel is definitively lost or the fault is temporary,

it holds its output and notifies the formation controller. The formation controller must know that is using held data because, although trajectory information can be held constant, absolute position (that is GPS data) can not, so it must interpolate position data using trajectory data.

The aircraft whose TX stops working must also be able to detect it because this is needed by the formation reconfiguration procedures. When, for any reason, one aircraft TX becomes not operational, that aircraft can no longer be a reference for the others. The TX fault has the effect of "deleting" all outgoing channels. From the point of view of Dijkstra's algorithm, the arcs leaving that node may be assigned a weight equal to infinity. These arcs will not be used in any optimal path where possible other connections exist, under the assumption that working arcs have positive finite weight. All the aircraft that used the faulty one as reference must reconfigure. From the standpoint of communication reconfiguration, the case of an aircraft loss corresponds exactly to the broken TX case.

3.3.1 Communication Topology Reconfiguration

After a failure, a fast reconfiguration procedure must be run to restore formation-keeping as quickly as possible. When an aircraft detects that its own RX is faulty, it must reconfigure the formation controller to use the VL information, if no connection to the VL is available, the aircraft cannot remain in the formation. When an aircraft detects a TX fault in its reference that is it loses its present communication channel, it must use a different node as reference. Dijkstra's algorithm is run again to find the new optimal set. Changing an incoming channel node affects that node's potential and the aircraft using it as a reference should change their reference as well. The failure event must be propagated to every aircraft of the formation that must run the

optimization algorithm even if their incoming channel is still functioning. A special communication channel, called Broadcast Channel is used to this end.

It is necessary to ensure that each run of the optimization algorithm leads to the same result in each aircraft. Since the implementation of Dijkstra's algorithm is deterministic, it is sufficient to ensure that each aircraft have the same information on present nodes occupation and on working TXs and RXs to guarantee that each separate run of the algorithm in all aircraft leads to the same solution. This will be also achieved using the Broadcast Channel.

After all nodes have completed the reconfiguration, the new graph is optimal again, and this procedure can then be repeated in case of successive failures without having to reconsider optimization of the whole graph. The procedure must be the fastest possible, because between the fault detection and the reconfiguration, the aircraft does not receive any trajectory information and the risk of conflict with other aircraft increases dramatically. Due mainly to this motivation, the reference channel reconfiguration procedure is run locally on each aircraft.

If one aircraft does not find an alternative communication path, it must leave the formation following a prescribed escape maneuver, which brings it safely outside even if the formation is maneuvering. Before taking the escape path, it must switch off its TX to prevent the aircraft that were using it as a reference to follow on the escape path. The basic idea beneath the safe escape procedure is that the aircraft is first brought to fly at a different altitude, then driven away from the estimated formation direction, and then driven back home if possible. Three different altitude levels: z-1, z-2 and z-3 apply depending on position inside the formation. Figure 3.4 shows possible escape maneuvers.

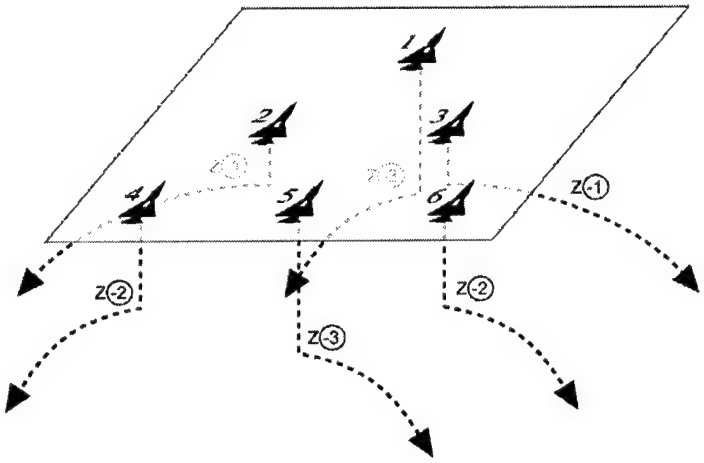


Figure 3.4: Safe Formation Escape Maneuvers

3.3.2 The Broadcast Channel

To keep all aircraft informed of what is happening to a specific node, and which of the nodes are active (that is nodes occupied by a vehicle), an additional broadcast communication channel is introduced that transmits data at low frequency; this new channel is called Broadcast Channel (BC). Communications on the broadcast channel are asynchronous because the formation must react to failure events with the shortest delay possible. At the same time, each aircraft notifies its presence in the formation with a periodic signal; if the communication period is Tactive seconds, all aircraft are updated on active aircraft changes at the most Tactive seconds after the fault. The information carried on the BC is in fact vital to the coordination of the formation as will be shown later on. In particular, without the BC, it could not be possible to inform all aircraft when and how to re-optimize the communications, to move aircraft

inside the formation without moving all the others using that one as reference and so on.

The use of BC communications is quite large during a post-fault recovery procedure but, BC communications are needed to coordinate the various aircraft and, although the carried information is very limited, they are vital for the propagation fault events and to avoid conflicting decisions. It must be stressed that no arbitration and no conflicting access may happen to the BC, unless two failures are simultaneous. While a fault is being serviced, other failures are kept in a priority queue and serviced sequentially. In this case, the aircraft subject to failure performs at least the communication channel re-optimization procedure to ensure safe, although sub-optimal, channel usage.

Failures in the BC transmitting and receiving devices were not considered in the present work.

3.4 Aircraft Position Reconfiguration

After the formation communications have been re-optimized, it may be necessary to move the aircraft inside the formation to fill holes left by a missing aircraft or to exchange two or more aircraft positions to reach desired formation geometry, and to maximize the formation-keeping capability and safety of all aircraft inside the formation.

Since formation safety and its precise control capability are measured by the total cost of the communication tree after a generic failure the new communication cost is greater than or equal to the preceding one. By moving and exchanging two or more aircraft inside the formation some arcs that were assigned a weight of infinity,

as unusable arcs because of broken TXs or RXs, could re-gain their original weight or have assigned a finite weight; thus it is possible to decrease the total communication tree cost.

The introduction of heuristic rules embedded into the reconfiguration process in terms of reconfiguration maps, accommodate this situation. Since the node-changing decision must be decentralized too, the algorithm that makes the decision must be deterministic in order to avoid simultaneous conflicting decisions by more than one aircraft in response to the same post-failure reconfiguration requirements. These rules constitute an expert system that decides, which is the best action to be taken after a failure:

- An aircraft with a broken RX and without a link to the VL must leave the formation because is unable of maintaining the formation.
- After the loss of an aircraft, the formation geometry must be brought to one of the desired geometries (See Figure 3.6).
- An aircraft with a broken RX has troubles keeping inter-aircraft distances because it has no knowledge of other aircraft positions, thus, the nearer to the leader it flies, the better it is.
- The formation leader, that is the aircraft in position 1, can lead the formation even with a broken RX.
- An aircraft with a broken RX, can be brought to lead the formation if the present leader has a working RX.
- An aircraft with a broken TX cannot be a reference for the others, and then its best position is at the back of the formation.

Based on the above, a number of reconfiguration maps was developed, which the reconfiguration manager in each aircraft applies in parallel to take post-fault decisions.

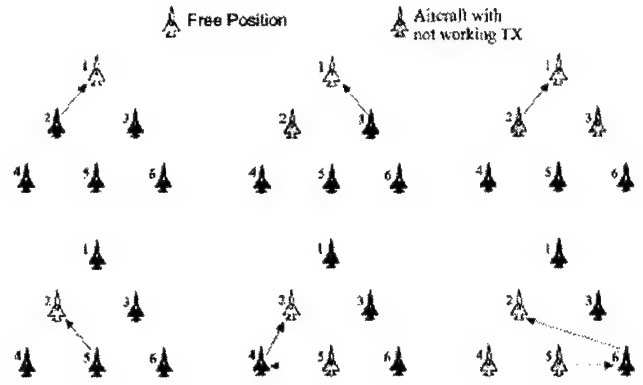


Figure 3.5: Reconfiguration Maps for 6 to 5 Aircraft

The use of BC is vital in this case; since it notifies each aircraft of the actions taken by the others.

3.4.1 Aircraft Loss

At the beginning of the formation mission, the communication optimization procedure is run to find the optimal communication scheme, but, when a node in the formation tree becomes free, the optimization procedure must be run again to find a new working communication channels set, just as in the case of TXs or RXs failures, but with the constraints given by the allowed geometries shown in Figure 3.2.

All the aircraft in the formation detect the event of an aircraft loss by listening to the BC. If after Tactive seconds one of the aircraft has not sent its "alive" signal through the BC, that aircraft is considered lost.

To describe the maneuvers needed to reach the new configuration, reconfiguration maps were used. The concept of RM was first introduced by the authors in Ref. 1, here their use has been extended to cover all the possible cases of broken TXs and

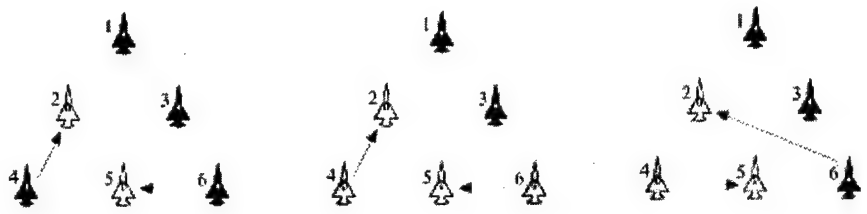


Figure 3.6: Reconfiguration Maps for 5 to 4 Aircraft

RXs in the formation. The RMs are grouped depending on the number of aircraft actually occupying the formation.

Figure 3.5 shows the RMs for the loss of one aircraft in a six aircraft formation. If position 1 (P1), the leader position, becomes empty, aircraft in position 2 (P2) takes its place, unless its TX is not working. In this case, aircraft in P3 moves to P1, but if its TX is not working the precedence to lead the formation goes to aircraft in P2. If P2 becomes free, aircraft in P5 takes its place, unless its TX is broken, in this case succeeds aircraft in P4, only if its TX is working, otherwise aircraft in P6 takes P2 and then the aircraft in P5 moves to fill the empty place. After this reconfiguration, if the aircraft in P1 has a nonfunctional TX, then the procedure in the next subsection is activated relative to the leader TX failure. The RMs for free position in the third row is not shown because it is trivial: if P4 or P6 become free, aircraft in P5 takes the free place. If the lost aircraft was in P5, no reconfiguration is necessary.

Figure 3.6 shows the RMs from 5 to 4 aircraft. If P1 is free the behavior is identical to the 6 to 5 RMs. If P2 becomes free, the aircraft in P4 or P6 are moved depending on which one has a working TX and with precedence to the aircraft in P4. The remaining aircraft completes the reconfiguration taking P5. If P3 becomes

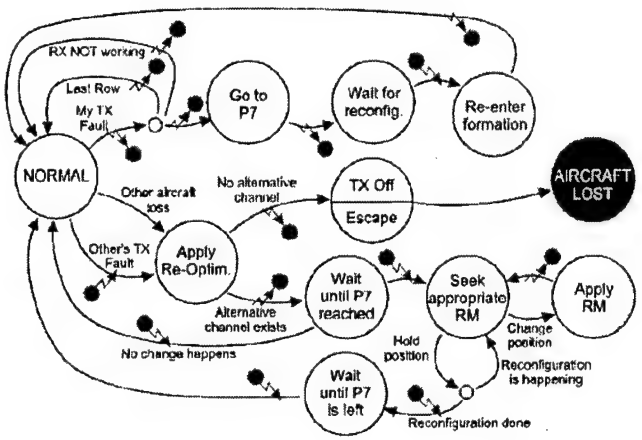


Figure 3.7: TX Failure Management Logic

free, the RMs are specular. If P4 or P6 become free, the remaining aircraft in the last row takes P5.

The RMs from 4 to 3 aircraft are not shown but are very intuitive: if P1 is free the behavior is identical to the 6 to 5 RMs. If P2 or P3 become free, the aircraft in P5 takes the free position. If P5 becomes free no reconfiguration happens. If the formation has 3 or two aircraft, RMs are trivial as well and behave as already described in the previous cases.

If a failure occurs in some aircraft TX, that aircraft is first moved inside the formation to become a leaf of the tree, where the TX capability is not needed. If the aircraft already occupies the last row or its RX is nonfunctional, no geometry re-configuration is needed. If a re-configuration is necessary, the faulty aircraft moves to a position outside the formation, just behind the last row, called position 7 (P7). The remaining aircraft reconfigure the geometry using the RMs as soon as P7 has been reached.

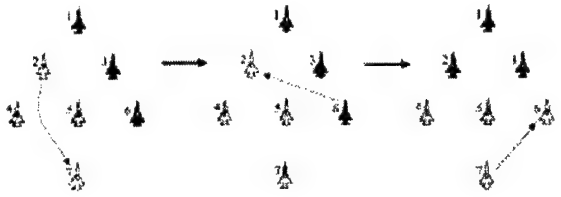


Figure 3.8: Example of reconfiguration after a TX Failure

When reconfiguration is done, the aircraft in P7 re-enters the formation in P5 if the total number of aircraft is 6 or 4, in P4 or P6 depending on the free one if the aircraft are 5, in P2 or P3 if the aircraft are 3 or 2. During these phases coordination is essential and it is achieved via the BC. Figure 3.7 shows a high-level finite state machine description of this process. The BC communications are represented by arrows entering or coming out from the BC, depending whether the corresponding event is generated by the state transition or it generates the transition respectively. The scheme shows also how an aircraft behaves if the re-optimization process produces no sub-optimal channels in alternative to the broken one. The state marked as Aircraft Lost is not a real state and it was introduced only to complete the machine description.

Figure 3.8 shows an example of a 6 aircraft formation, with the aircraft in P4 and P5 having broken TXs. When the aircraft in P2 loses its TX, it moves to position P7, the aircraft in P6 takes its position, the aircraft in P5 should take P6 if this was a reconfiguration after a real aircraft loss but the Reconfiguration Manager knows that the aircraft in P7 is going to re-enter the formation, thus, to avoid useless position changes, leaves P6 free and drives the aircraft in P7 to move to P6.

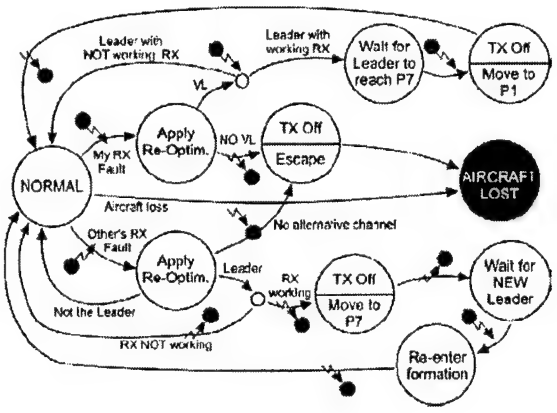


Figure 3.9: RX Failure Management Logic

3.4.2 Receiver (RX) Failure

After an aircraft detects a failure in its RX device, it either switches to use the VL trajectory, or it must leave the formation. As explained before, an aircraft with a faulty RX but with a virtual connection to the VL can lead the formation without affecting general performance. Thus, if the current leader has an operational RX, the two aircraft can change their position. First the leader moves to P7, the faulty aircraft takes P1, then the leader re-enters the formation in the place left free by the other. All other aircraft hold their positions during this phase. The finite state machine in Figure 3.9 describes this logic.

The leader, before moving to P7 switches its TX off to avoid being followed by other aircraft. It will switch it on again after being in its new position inside the formation and back to the "Normal" state. The faulty aircraft, before moving to P1, switches its TX off for the same reason. It will turn it on again when has started to lead the formation.

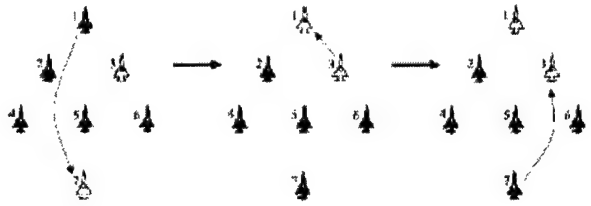


Figure 3.10: Example of Reconfiguration after RX Failure

Figure 3.10 shows an example with a 6 aircraft formation. When the aircraft in P3 loses RX capability, the leader moves to P7, the aircraft in P3 moves to P1, then the former leader takes P3 and reconfiguration is ended. In this case the formation keeping capability is unaltered and the communication optimal solution is the same as before failure. Note that in the case of a second RX failure (as shown in Figure 3.9), the faulty aircraft remains in the same position, if connected to the VL, otherwise it must exit the formation.

3.4.3 Example: Failure Management during Aircraft Loss

As an example of formation management procedure, we consider the case of loss of aircraft in a 6-ship formation. As shown in Figure ??, the manager is present in each aircraft and handles the commands to the autopilots based on the information received, transmitted, and communications from the broadcast channel. The schematic structure is shown in Figure 3.11.

The status of the aircraft is processed by two Finite State Machines (FSM) defined as control's panel, and fault's manager. The control's panel has three parallel superstates (Receiver, Transmitter, Aircraft_Status), it contains four events (one local

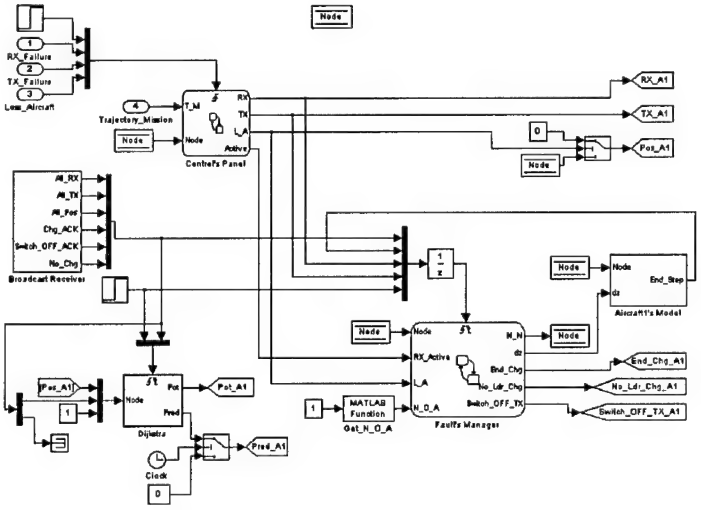


Figure 3.11: Simulink Diagram of Formation Manager

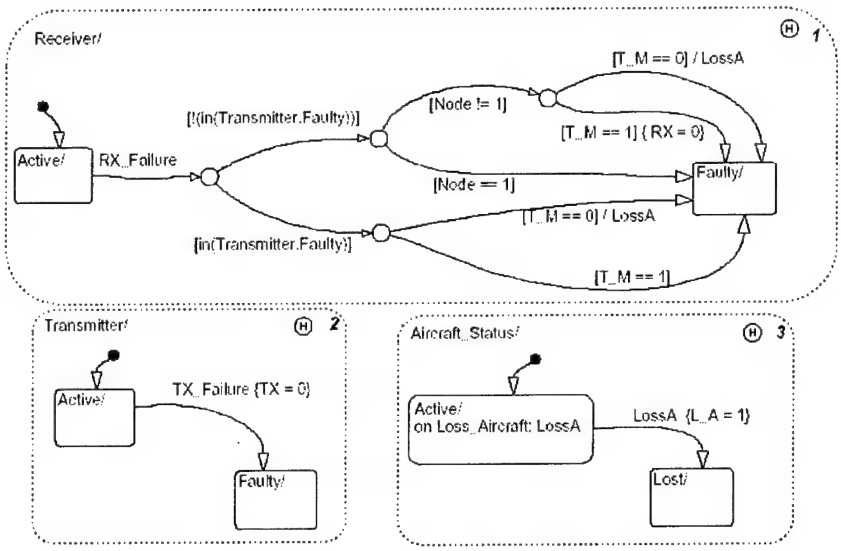


Figure 3.12: Control's Panel Diagram

and three as input), and has six datasets (two inputs and four outputs). Figure 3.12 shows the stateflow® implementation of the parts of Figures 3.7 and 3.9 relative to the control's panel FSM.

Let us describe now the working procedure of the formation's manager during the loss of aircraft 3 in a formation of six, with aircraft 5 replacing it. Figure 3.14 shows the graph flow with node potentials and arc weights values, in the nominal situation.

The fault's manager FSM has a single superstate containing seven mutually exclusive states (Nothing_Happens, Lost_Aircraft, My_RX_Faulty, Leader, RM, My_TX_Faulty, TX_Faulty). The stateflow representation is shown in Figure 3.13. When aircraft 3 is lost, each vehicle runs the algorithm again setting to infinity the weights of the arcs entering and exiting node 3. The result is shown in Figure 3.15.

As far as aircraft 3 is concerned, its control's panel activates the state Air-

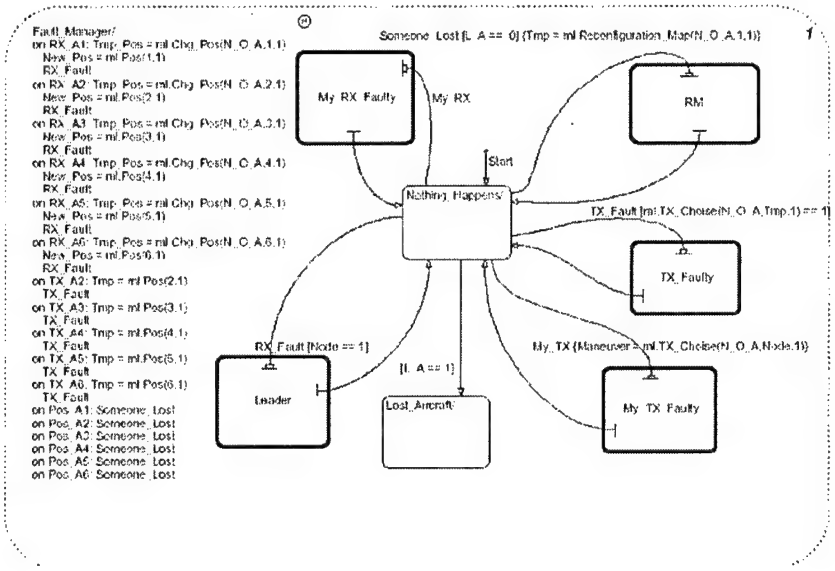


Figure 3.13: Fault's Manager Diagram

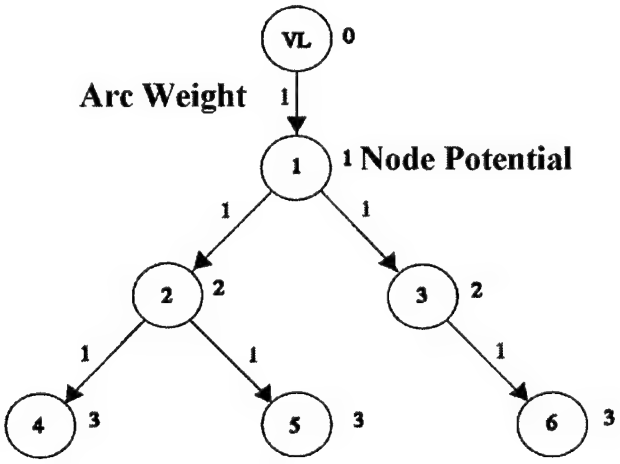


Figure 3.14: Minimum Cost Graph from Dijkstra's Algorithm, Nominal

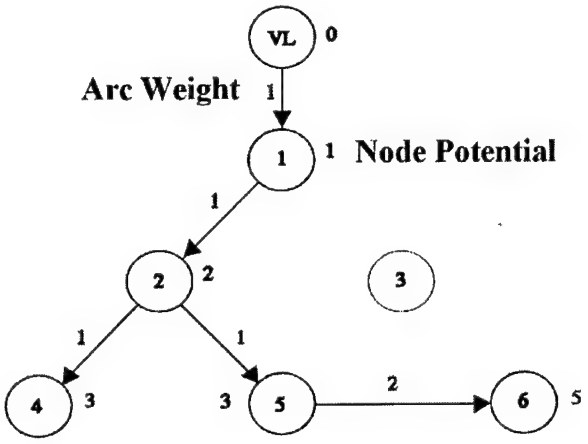


Figure 3.15: Minimum Cost Graph from Dijkstra's Algorithm, ignoring Node 3

craft_Status_Lost and the value of 1 is assigned to the output variable L.A (Lost aircraft). The latter causes the event Pos_A3 in all the other aircraft fault's managers, via the broadcast receivers, in addition of activating Dijkstra's algorithm. Event Pos_A3 forces the transition from state Nothing_Happens to state Lost_Aircraft in the Fault's Manager of aircraft 3, since L.A.= =1 (see Figure 3.16).

The other aircraft in the formation, receive event Pos_A3, and execute the reconfiguration maps, going to state *RM* (see top right block in Figure 3.17). The Fault's Manager in aircraft 5, in particular, evaluates the function Reconfiguration_Map obtaining the value 3 as new desired position in the formation, the transition is valid and the state *RM* is activated (see Figure 3.17). Aircraft 5 understands that it must change position to node 3, shuts down the *TX* (see top left block in Figure 3.18), waits for the others to perform channel reconfiguration (Switch_OFF_ACK), and then executes the maneuver to go to the final destination. In this case, two types of

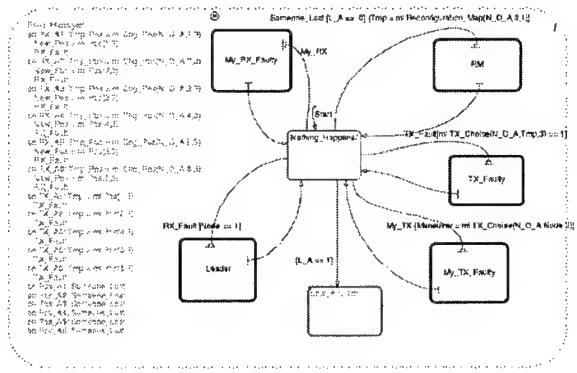


Figure 3.16: Fault's Manager Aircraft 3 after its Loss

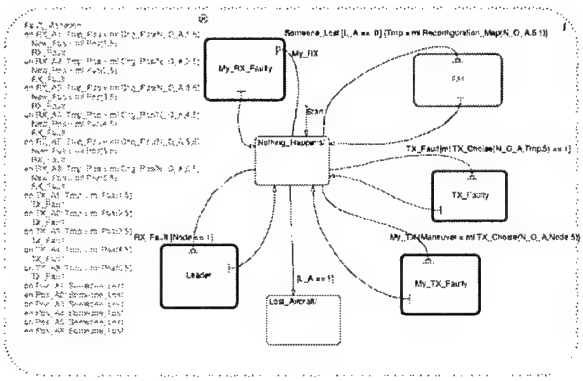


Figure 3.17: Fault's Manager Aircraft 5 after Loss of Aircraft 3

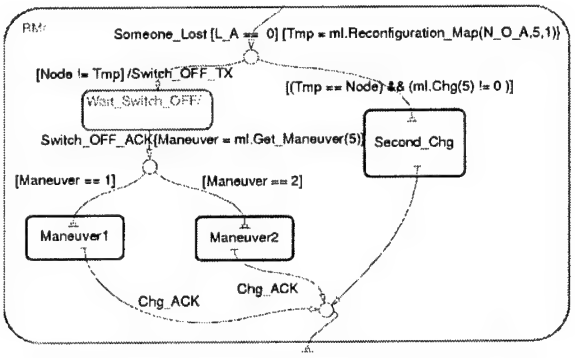


Figure 3.18: RM Aircraft 5 TX Shutdown

maneuvers are available as shown in Figure 3.18, Maneuver1 and Maneuver2. The former is achieved with a change in altitude, the latter without. The choice is based on safety reasons. The complete sequence is described then by the sequence of Figures 3.17, 3.18 and 3.19.

When aircraft 5 has decided to move to Node 3, the shutdown of its TX activates Dijkstra's algorithm, setting the weight to infinity for the arcs leaving Node 5 (this avoids other aircraft to follow 5 during the reconfiguration maneuver). The result is shown in Figure 3.20. The new optimal graph, with aircraft 5 in the new position is computed, as in Figure 3.21.

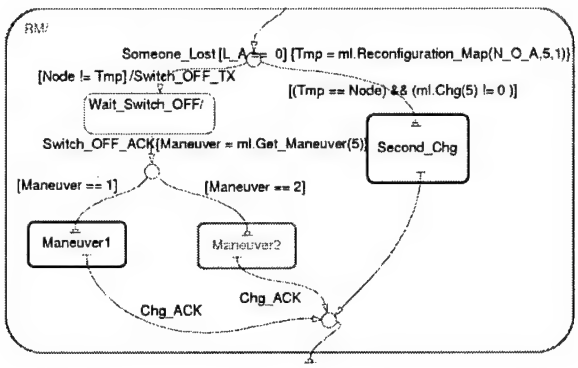


Figure 3.19: RM Aircraft 5 TX Maneuver to Node 3

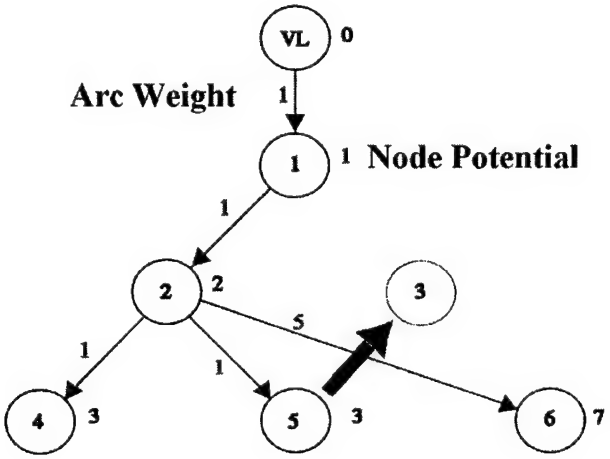


Figure 3.20: Minimum Cost Graph ignoring Node 3, and all Arcs exiting Node 5

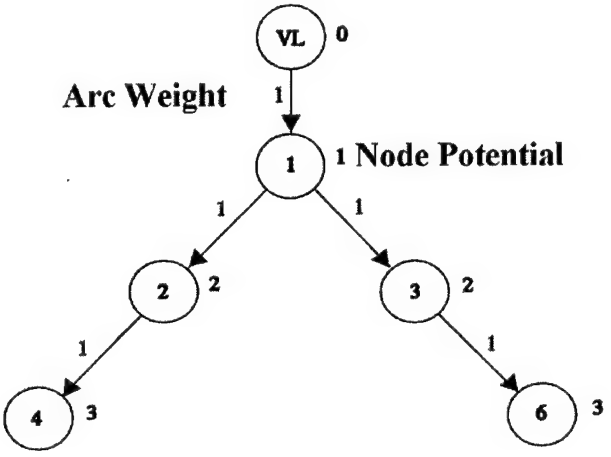


Figure 3.21: Final Minimum Cost Graph

Part II

Guidance and Management of Flight Vehicles

Chapter 4

Stability of Gain Scheduling using Fuzzy Sets

4.1 Introduction

This chapter and the next describe the research performed in the area of stability and guidance of autonomous flight vehicle formation.

Since its first appearance, Takagi-Sugeno (TS) fuzzy model theory [Takagi, 1985] has proven useful in the description of nonlinear dynamic systems as a means of blending of models obtained by local analysis. Such descriptions are referred to as model based fuzzy systems (MBFS). In addition, the TS approach can be used for the synthesis of fuzzy gain-scheduled controllers. The stability of MBFS was studied by Hallendorn, Palm and Driankov\cite [Hallendorn, 1996a], [Hallendorn, 1996b], who defined a stability test by imposing some conditions on the local control laws. The present work describes a new stability criterion, which relaxes the bounds in

[Hallendorn, 1996b], yielding a less conservative condition. Two case studies are presented comparing the use of off-equilibrium versus equilibrium grid points, and fuzzy versus crisp scheduling.

4.2 Modeling and Control

Consider a nonlinear continuous and continuously differentiable system of the form:

$$\dot{\bar{x}} = f(\bar{x}, \bar{u}) \tag{4.1}$$

where$\bar{x} \in \Re^n, \bar{u} \in \Re^m$, and $\Re^n \times \Re^m \rightarrow \Re^n$. We wish to design a controller capable of following some desired trajectory , where $\bar{x}_r$ is a differentiable, slowly varying state trajectory, and $\bar{u}_r$ is the nominal input necessary to follow the unperturbed $\bar{x}_r$ state[2,3]. Let us define a subset $XU \subset \Re^{n+m}$ of the system's state and input spaces as a bound on all the possible state and input values. Let us also define a set of operating points as $(x_i, u_i) \in XU, i \in I$ with I set of all positive integers that form a regular (or irregular) grid J in the trajectory space. Linearization of Eq. (4.1) about all the points in J yields:

$$\begin{cases} A_i = \frac{\partial f}{\partial x}\Big|_{(x_i,u_i)} \\ B_i = \frac{\partial f}{\partial u}\Big|_{(x_i,u_i)} \end{cases} \tag{4.2}$$

resulting in perturbed dynamics about the linearization points given by:

$$\begin{cases} \dot{\bar{x}} = A_i(\bar{x} - \bar{x}_i) + B_i(\bar{u} - \bar{u}_i) + f(\bar{x}_i, \bar{u}_i) = A_i\bar{x} + B_i\bar{u} + \bar{d}_i \\ \bar{d}_i = f(\bar{x}_i, \bar{u}_i) - A_i\bar{x}_i - B_i\bar{u}_i \end{cases} \tag{4.3}$$

When the linearized systems (4.3) are interpolated through a TS model, a nonlinear approximation of (4.1) is obtained, given by:

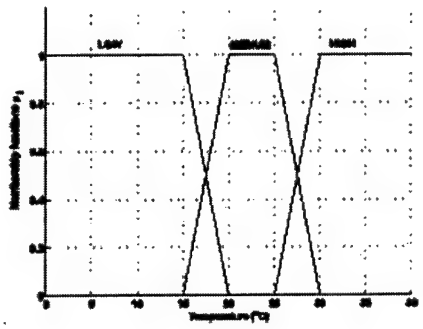


Figure 4.1: Example of Fuzzy Rule Set

$$\dot{\bar{x}} \cong \hat{f}\left(\bar{x},\bar{u}\right) = \sum_{i\in I} \mu_i\left(\bar{x},\bar{u}\right) \cdot \left(A_i\bar{x} + B_i\bar{u} + \bar{d}_i\right) \tag{4.4}$$

A fuzzy control law for system (4.4) is also designed as a gain-scheduling controller based on a TS model. Under the hypothesis of controllability for all system matrices pairs and being all the states measured, full state feedback linear control laws can be synthesized, and interpolated through a fuzzy TS system yielding:

$$\begin{aligned} \overline{u} &= \overline{u}_r + \sum^{j\in I} \upsilon_j\left(\bar{x},\bar{u}\right) \cdot K_j\left[\left(\bar{x} - \bar{x}_j\right) - \left(\bar{x}_r - \bar{x}_j\right)\right] \\ &= \overline{u}_r + \sum^{j\in I} \upsilon_j\left(\bar{x},\bar{u}\right) \cdot K_j\left[\left(\bar{x} - \bar{x}_r\right)\right] \end{aligned} \tag{4.5}$$

In Eqs. (4.4) and (4.5), the expressions μ_i, υ_j represent the TS linear membership functions relating the input variables to the fuzzy domain described by IF-THEN-ELSE rules consequent. The fuzzy system membership functions are chosen such as they constitute a convex sum over the input range XU. An example is shown in Figure 4.1

Substituting Eq. (4.5) in Eq. (4.4), the closed loop perturbed system dynamics

become:

$$\begin{cases} \dot{\bar{x}} - \dot{\bar{x}}_r = \sum_i \mu_i(\bar{x}, \bar{u}) \cdot \left[A_i + B_i \left(\sum_j v_j(\bar{x}, \bar{u}) K_j \right) \right] (\bar{x} - \bar{x}_r) + \varepsilon \end{cases} \tag{4.6}$$

$$\varepsilon = \sum_i \mu_i(\bar{x}, \bar{u}) \cdot \left(A_i \bar{x}_r + B_i \bar{u}_r + \bar{d}_i \right) - \dot{\bar{x}}_r \tag{4.7}$$

Note that the term $\sum_i A_i \bar{x}_r$ is added and subtracted so that the matrix

$$\sum_i \mu_i(\bar{x}, \bar{u}) \cdot \left[A_i + B_i \left(\sum_j v_j(\bar{x}, \bar{u}) K_j \right) \right] \tag{4.8}$$

gives the dynamics of the perturbation from the desired trajectory. Also, from the definition of $\bar{d}_i$, ε represents the error with respect to $\bar{x}_r$ due to the approximation of f with the TS model.

4.3 Stability Analysis

Let us now derive the asymptotic stability conditions of the TS fuzzy gain-scheduling controller around $(\bar{x}_r, \bar{u}_r)$.

Definition: *Given the grid point set J and any linearized dynamics $(A_i, B_i), i \in J, J_i$ is defined as the set of all indexes m of the neighbourhood points of $(\bar{x}_i, \bar{u}_i)$, whose controllers K_m have a non negligible influence over(A_i, B_i).*

>From the above, J_i contains all points m such that $v_m(\bar{x}, \bar{u}) > 0$, $\forall (\bar{x}, \bar{u}) \in \{(\bar{x}, \bar{u}) : \mu_i(\bar{x}, \bar{u}) > 0\}$.

Given a generic input state pair $(\bar{x}_l, \bar{u}_l), l \notin I$, the stability property for the tracking error $(\bar{x}_l - \bar{x}_r) \rightarrow 0$ requires that the following conditions be satisfied:

Condition 1: *Suppose* $(\bar{x}_i, \bar{u}_i) \in J$ *is the nearest linearization grid point to the operating point* $(\bar{x}_l, \bar{u}_l) \in J$. *The system* (A_i, B_i) *remains closed-loop stable using a convex combination of controllers* $K_m, m \in J_i$.

Condition 1 is verified using the following test, which is based on differential inclusion theory [Boyd, 1994], [Marullo, 2001] the test guarantees that the TS modelling of the plant $\dot{\bar{x}} = f(\cdot)$ is stable when controlled by the TS fuzzy controller for all possible controller combinations. Consider the closed-loop system dynamics about $(\bar{x}_i, \bar{u}_i)$:

$$\dot{\bar{x}} = \left[A_i + B_i \left(\sum_j v_j(\bar{x}, \bar{u}) K_j \right) \right] \bar{x} \tag{4.9}$$

$$= \sum_j v_j(\bar{x}, \bar{u}) \cdot [A_i + B_i K_j] \bar{x} \tag{4.10}$$

obtained by a convex combination of controllers$K_m, m \in J_i$.

Equation (4.9) has the form of a polytopic differential inclusion, where the vertices are the matrices $A_i + B_i K_j, i \in J_i$.Differential inclusion theory states that closed-loop stability for the vertices of a polytope yields stability of the whole convex combination, therefore stability of all $A_i + B_i K_j, i \in J_i$is required. The stability test is repeated for all grid points obtaining:

$$\forall i \in I, \forall j \in J_i, \exists P_i > 0 : (A_i + B_i K_j)^T P_i + P_i (A_i + B_i K_j) < 0 \tag{4.11}$$

Inequality (4.11) can be easily solved using Linear Matrix Inequalities (LMI) techniques. If the LMI test fails, then the grid J must be made denser. Furthermore, the LMI test suggests where to add additional linearization points, in order to make the closed-loop system stable. The proposed stability test improves that of

[Khargonekar, 1987]; since it does not require that the closed loop eigenvalues be the same for all operating points $i \in I$ ([Khargonekar, 1987], Equation (23)).

Condition 2: *The approximation error due to linearization and successive TS fuzzy modelling with respect to the original nonlinear system is small enough as not to compromise robust stability with respect to structured uncertainties.*

Let us suppose that the desired closed-loop dynamics are given by:

$$A_d = A_i + B_i K_i, \forall i \in I \tag{4.12}$$

then, from Eq. (4.9):

$$\sum_j v_j(\bar{x}, \bar{u}) \cdot [A_i + B_i K_j] \bar{x} = A_d + \sum_j v_j(\bar{x}, \bar{u}) \delta A_{ij} \tag{4.13}$$

with:

$$\delta A_{ij} = B(\bar{x}_i, \bar{u}_i) [K(\bar{x}_j, \bar{u}_j) - K(\bar{x}_i, \bar{u}_i)] \tag{4.14}$$

[Hallendorn, 1996a], [Marullo, 2001] propose to test the stability of

$$A + \sum_i \mu_i(\bar{x}, \bar{u}) \sum_j v_j(\bar{x}, \bar{u}) \cdot \delta A_{ij} \tag{4.15}$$

using the robust stability theorem under structured uncertainties found in [Khargonekar, 1987]. In order to satisfy Condition 2, the uncertainty in the systems (A_i, B_i) due to linearization errors is modeled by a set of possible parametric uncertainties. The maximum parametric variations for which stability is guaranteed can be computed using LMI techniques. If this result is larger than the maximum functional error, then the system remains stable in the entire convex combination of grid J points. The functional approximation error due to linearization is computed as follows [Johansen, 1993]:

$$\left\| f - \overrightarrow{\to} f \right\|_{\infty} \leqslant \frac{M}{2} \left[\Delta \left(J, XU \right) \right]^2 = \varepsilon \tag{4.16}$$

where ε is the maximum approximation error, and

$$\begin{gathered} \left\| \nabla^2 f \left(\bar{x}, \bar{u} \right) \right\|_{\infty} \leqslant M, \forall \left(\bar{x}, \bar{u} \right) \in XU \\ \Delta \left(A, B \right) = \inf_{a \in A} \sup_{b \in B} \left\| a - b \right\|_2 \end{gathered} \tag{4.17}$$

The term $\Delta \left(J, XU \right)$ represents a measure of the inverse grid point's density: a small Δ implies a dense grid. In fact, the particular choice for M gives a conservative estimate for ε. Some regions of XU may be more "regular", thus requiring a sparse grid. A local value ε_i for the error can be defined as:

$$\frac{M_i}{2} \left[\Delta \left(J_i, conv \left(\left(\bar{x}_k, \bar{u}_k \right), k \in J_i \right) \right) \right]^2 = \varepsilon_i \tag{4.18}$$

$$\left\| \nabla^2 f \left(\bar{x}, \bar{u} \right) \right\|_{\infty} \leqslant M_i, \forall \left(\bar{x}, \bar{u} \right) \in conv \left(\left(\bar{x}_k, \bar{u}_k \right), k \in J_i \right) \tag{4.19}$$

where $conv(.)$ indicates convex closure. Grid density can be adapted depending on the variousM_i, to keep ε_i below the desired approximation error ε. To test the robustness of the control system, let us define a set of structured uncertainties $E_{l,m}$ as matrices with 1 in the (l, m) position and zero otherwise. Define also the finite set:

$$M_{Ji} = \left\{ (l, m) : \left(A_{\ i} + B_i K_j \right)_{(l,m)} \neq 0, j \in J_i \right\} \tag{4.20}$$

and an index p to the elements in the set M_{Ji}. A matrix $E_{l,m}$ exists for each element p in M_{Ji}.Thus, the generic structured uncertainty can be modeled as $E_p = k_p E_{l,m}$ where the scalar k_p is the magnitude of uncertainty, and p the index in M_{Ji} corresponding

to the given pair (l, m). Note that a matrix $E_{l,m}$ exists for each element in M_{Ji}. We can then solve the following LMI problem in the unknowns P_i and k_p:

$$\forall i, \forall j \in J_i \begin{cases} \rho_i = \max \sum_{p=1}^{|M_{Ji}|} |k_p| \\ P_i > 0 \\ (A_i + B_i K_j)^T P_i + P_i (A_i + B_i K_j) + \\ + \left(\sum_{p=1}^{|M_{Ji}|} E_p\right)^T P_i + P_i \left(\sum_{p=1}^{|M_{Ji}|} E_p\right) < 0 \end{cases} \tag{4.21}$$

yielding the bound ρ_i to the maximum allowable uncertainty.

If $\rho_i > \varepsilon_i$, $\forall i$, then system (4.1) with the fuzzy gain-scheduling controller given by (4.5) is stable for all trajectories in XU. If the stability test fails, it is necessary to select a denser grid. The values of ρ_i that break the ε_i threshold indicate which regions in XU require a denser grid.

Suppose now that (4.21) yields $\rho_i < \varepsilon_i$, we must find a new approximation error ε_i' such that $\rho_i' > \varepsilon_i'$. This implies that M_i and $\Delta(\cdot)$ must be reduced by increasing the number of grid points. The additional grid points must belong to $conv((\bar{x}_k, \bar{u}_k), k \in J_i)$. The new grid J' satisfies therefore $J' \supset J$. The following now holds:

$$conv((\bar{x}_k, \bar{u}_k), k \in J_i') \subset conv((\bar{x}_k, \bar{u}_k), k \in J_i) \tag{4.22}$$

>From (4.19) we have:

$$\begin{cases} M_i' \leqslant M_i \\ \Delta(J_i', conv((\bar{x}_k, \bar{u}_k), k \in J_i')) < \Delta(J_i, conv((\bar{x}_k, \bar{u}_k), k \in J_i)) \end{cases} \tag{4.23}$$

So from (4.18) $\varepsilon_i' < \varepsilon_i$. LMI optimization is run again over J', yielding ρ_i'. The procedure is repeated until $\rho_i' > \varepsilon_i', \forall i$.

4.4 Case Studies

The first case study is a classical nonlinear system benchmark. Consider the system given by:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = x_1^2 + x_2^2 + u \end{cases} \tag{4.24}$$

The objective of the control design is to follow reference trajectory given by a step command on the first state, i.e. $\bar{x}_r = \begin{bmatrix} A & 0 \end{bmatrix}^T$. A pole placement method is used to assign the closed-loop linearized spectrum to (-2.5, -5.0). Two fuzzy controllers were designed:

Conventional controller case: this controller is obtained considering equilibrium points only. The equilibrium points for (4.24) belong to the manifold

$$\left\{ (x_{1e}, x_{2e}, u_e) | \, x_{2e} = 0, u_e = -(x_{1e})^2 \right\}$$

from which the following were selected: [(0.5,0,-0.25), (1.5,0,-2.25), (2.5,0,-6.25), (3.5,0,-12.25)],

Off-equilibrium case: the following linearization points were selected: [(0.5,-1,0), (1.5,-1,0), (2.5,1,0), (3.5,1,0), (0.5,0,0),(1.5,0,0), (2.5,0,0), (3.5,0,0), (0.5,1,0), (1.5,1,0), (2.5,1,0), (3.5,1,0), (0.5,4,0), (1.5,4,0), (2.5,4,0), (3.5,4,0)]. Note that this set contains non-equilibrium points as well.

Figure 4.2 shows the controlled system's response to a step of amplitudes 3 and 4.

Figure 2: Step Response

It is evident from the time histories that the off-equilibrium solution outperforms the conventional one. The conventional fuzzy controller can not track the command

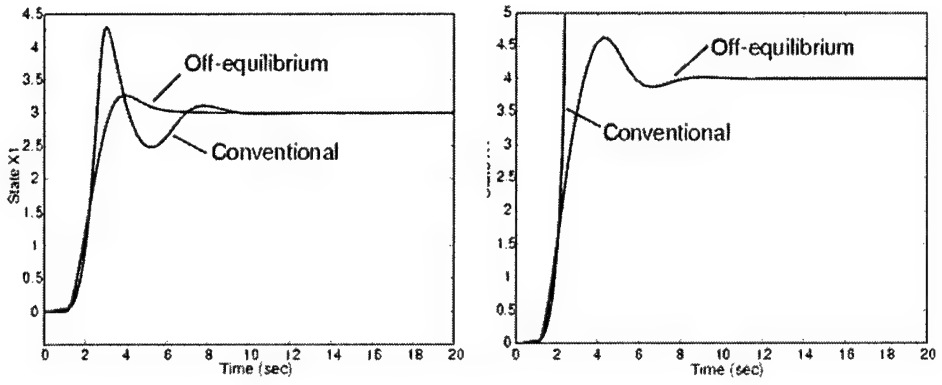


Figure 4.2: Step Response

with amplitude 4, while the off-equilibrium fuzzy controller tracks the command with fast settling time and little overshoot. Although the final state (4,0) belongs to the stability manifold, the conventional controller can not reach it in response to a step command because its state crosses regions where $x_2 >> 0$,which is too far from the equilibrium manifold. The off-equilibrium controller succeeds in tracking the command because it was designed to be stable in a larger region XU containing all the state trajectories. From Condition 1, we have that the convex combination of fuzzy controllers in the intermediate region retains stability.

Robustness to approximation error is also satisfied. In the calculation of course, the structured uncertainty matrices are zeroed for the dynamic matrix entries relative to linear functions (in such a case even if the Jacobian matrix entry is different from zero, the approximation error results null). That is if the $x_1$derivative were nonlinear, the approximation test would fail suggesting a necessity for a finer grid (the Johansen index in the proposed example depends mainly on the grid step $\Delta\left(J_i, conv\left(\left(\bar{x}_i, \bar{u}_i\right), i \in J_i\right)\right)$.

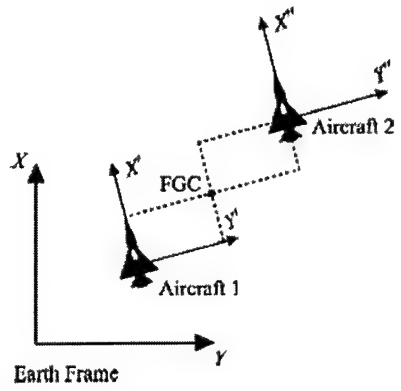


Figure 4.3: Schematic of a Two-Ship Formation

The second case study is a formation control problem for unmanned aircraft executing tight turns as depicted in Figure 4.3.

The dynamics of the aircraft are based on a standard point mass model with speed, flight path angle, and heading angle as states, and thrust, bank angle and load factor as inputs. The formation error is computed as distances of leader and wingman from the formation geometric center (FGC) defined in [Blake, W., 2000], where complete modelling and autopilot design are presented. The formation controller is limited to planar motion with a constant speed of 20 m/sec [Giulietti, F. *et al.*, 2000], and the only variation on the heading angle $\chi \in \left(-\frac{\pi}{4}, \frac{\pi}{4}\right)$, this implies $XU = [-\pi, \pi] \times [-\pi, \pi]$. Nine operating points were used to form the grid, and for each point a standard LQ-Servo was designed using a LMI procedure[4]. Performance comparison was made between TS fuzzy gain scheduling, and a crisp schedule [Khargonekar, 1987], [Nichols, 1993].

The stability test described in the previous section was successful, as well as the

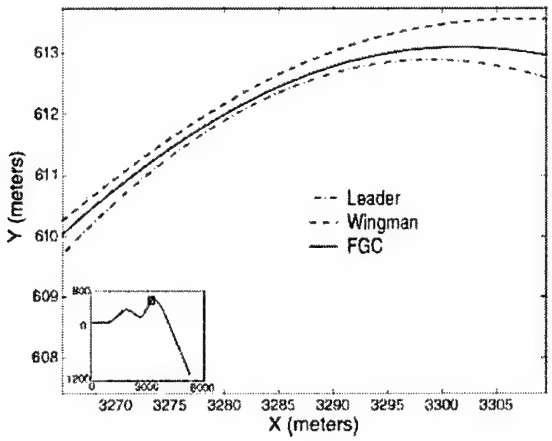


Figure 4.4: Heading Change tracking

test for the approximation error ($\varepsilon = 7.969, \Delta = \pi/8, M = 19.6$). A comparison was made between the conventional controller (rigid gain scheduling with no mixing of gains), and the fuzzy gain-scheduled system (FGS). The validation and comparison was performed by testing for a sequence of tight turns for which the crisp scheduled controller becomes unstable because is more sensitive to high frequency changes in the scheduling, while the fuzzy one does not. Results of the performance of the FGS are shown in Figures 4.4 and 4.5.

In Figure 4.4 we can see a detailed section of the trajectory indicating the closeness of the formation and the position of the aircraft relative to the FGC. Figure 4.5 shows the behaviour of the X-Y distances with respect to the formation geometry center, with the nominal (desired) values set at 5 meters for each component.

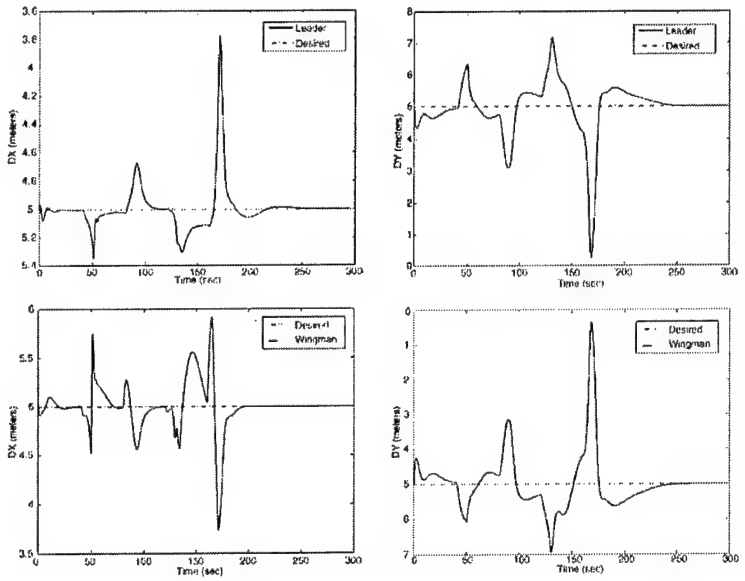


Figure 4.5: Horizontal Distances to FGC

Chapter 5

A Guidance Methodology using Fuzzy Sets and Waypoints

Chapter 6

Fuzzy Guidance

6.1 Introduction

This section describes the development, the simulations and the first results achieved of an innovative guidance scheme based on fuzzy systems. The aim of such a guidance system is having a tool for easy guidance of a single vehicle, groups or formations through a set of waypoints generated by a mission planning algorithm.

The guidance system requirements are:

1. capability to pass a set of waypoints in the prescribed order.
2. Possibility to specify the desired crossing velocity and heading for each waypoint.
3. To handle waypoints set where successive waypoints may not necessarily be on directly flyable routes (e.g. two waypoints one above the other).
4. Capability to reconfigure quickly the waypoint set without delay; possibly in response to some change in the mission scenario.

5. To reach fixed waypoints as well as to track and reach waypoints that are moving with relatively low speed and acceleration.

The Fuzzy Guidance System (FGS) assumes that the vehicle knows its position and targets/waypoints position as well as their velocity vector, in the case they are moving.

In generic waypoints are described in a 5-dimensional space: position in 3-dimensions plus desired crossing heading and velocity. In the case of waypoints that model moving targets, the desired crossing velocity is meant as relative velocity between the vehicle and the target .

For guidance study, the generic aircraft is assumed to be autopiloted in velocity, heading and flight path angle. The FGS is based on various standard Takagi-Sugeno (T-S) [Takagi, 1985] fuzzy controllers that generate separately velocity, heading and flight path angle references for the autopilots. The choice of the fuzzy systems arises from the need to specify desired waypoint's crossing direction, since traditional proportional guidance [K. R. Lee and Al., 2000] techniques do not allow specifying it. T-S, among others motivations, have been chosen because parts of the control algorithms need to perform surface approximation that must be fast from a computational standpoint.

6.2 Aircraft Dynamics and Control

The aircraft guidance problem is addressed, by designing an inner nonlinear control loop first, which allows tracking of commanded velocity (V_d), flight path (γ_d) and heading (χ_d). Then, an outer loop, that is the actual fuzzy guidance system (FGS), generates a reference path command in terms of desired velocity, flight path and

heading for the inner loop, in order to reach the desired waypoint.

6.2.1 Aircraft Dynamics

For the purpose of our study, a mathematical model of the aircraft was developed following some basic assumptions:

- Flat, non rotating earth.
- Standard atmosphere, no wind effects.
- There are no side-slip forces and the β angle is always zero.
- The aircraft's movements around its center of mass (attitude) can be neglected.

Under these assumptions a three degrees-of-freedom point-mass mathematical model is used. The model consists of three first order non-linear differential equations, in spherical coordinates:

$$\dot{V} = \frac{(T - D)}{m} - g \sin\gamma \tag{6.1}$$

$$\dot{\gamma} = \frac{g}{V}(n \cos\phi - \cos\gamma) \tag{6.2}$$

$$\dot{\chi} = \frac{g\, n \sin\phi}{V cos\gamma} \tag{6.3}$$

The state variables that describe the aircraft motion are: airspeed (V), flight path angle (γ) and heading angle (χ), and the input variables are thrust (T), load factor (n) and bank angle(ϕ), further model parameters are the aerodynamic drag (D) and the aircraft weight (W).

The complete aircraft dynamics are summarized as

$$\dot{x} = h(x, u) \tag{6.4}$$

where $x = [V, \gamma, \chi]$ and $u = [T, n, \phi]$.

6.2.2 Aircraft Control System

Assuming perfect modeling, the aircraft dynamics as in equation (6.4) can be feedback linearized with the following control laws:

$$T = k_v(V_d - V)m + mg\sin\gamma + D \tag{6.5}$$

$$n\cos\phi = \frac{V}{g}\left[k_\gamma\left(\gamma_d - \gamma\right) + \cos\gamma\right] = c_1 \tag{6.6}$$

$$n\sin\phi = \frac{V}{g}k_\chi\left(\chi_d - \chi\right)\cos\gamma = c_2 \tag{6.7}$$

from (6.6) and (6.7) n and ϕ are given by:

$$n = \sqrt{{c_1}^2 + {c_2}^2} \tag{6.8}$$

and

$$\phi = tan^{-1}\left(\frac{c_2}{c_1}\right) \tag{6.9}$$

with K_V, K_γ and K_χ positive constants, and V_d, γ_d and χ_d the desired state trajectories.

The resulting linear system is:

$$\begin{cases} \dot{V} = K_V\left(V_d - V\right) \\ \dot{\gamma} = K_\gamma\left(\gamma_d - \gamma\right) \\ \dot{\chi} = K_\chi\left(\chi_d - \chi\right) \end{cases} \tag{6.10}$$

Thus, a complete state variable decoupling can be achieved. In the case of inexact modeling, decoupling and linear behavior is no more guaranteed.

6.3 Fuzzy Guidance

The design of the FGS had various revisions that can be summarized in three FGS that are succesive in time and increasing in performances and features respect to guidance requirements. The first FGS accomplishes the first four requirements (1 to 4) but is not capable to track moving waypoints. In the second FGS, the structure of horizontal guidance changes greatly, two controllers are employed for short and long distance guidance; the performances of the first one are improved, some singularities in the control law are removed and the number of fuzzy rules is reduced. The third FGS achieves the 5th requirement: the capability to follow a moving waypoint and cross it with the desired crossing heading and relative velocity.

6.3.1 First FGS Design

Two components constitute the guidance system: the Waypoint Generator (WG) and the Fuzzy Guidance System (FGS). The desired trajectory is specified in terms of a list of waypoints without any requirement on the path between two successive waypoints. A waypoint is given in cartesian-space coordinates (X_W, Y_W, H_W) and a desired crossing speed (V_W) and heading angle (χ_W) are used to obtain a preferred approaching direction and velocity, thus the waypoint belongs to a five-dimensional space $W = [X_W, Y_W, H_W, V_W, \chi_W]$. The WG holds a list of waypoints (WL) in 5-D, checks aircraft position, and updates the desired waypoint when the previous one has been reached within a given tolerance. When all waypoints have been reached, it holds the last one so that the aircraft loops around it. Otherwise it could be possible to select the starting point as the last waypoint. The waypoint generator's only task is to present the actual waypoint to the FGS. At this point of this research, no dead-

reckoning or navigational errors are modeled so the WG and the FGS know the exact aircraft positions, velocity and heading.

Between the WG and the FGS, a coordinate rotation system transforms earth-fixed-frame position errors into waypoint-frame relative errors. Each waypoint defines a coordinate frame centered in the waypoint position (X_W, Y_W, H_W) and rotated by χ_W around the H-axis. This coordinate transformation allows to synthesize a fuzzy rule-set valid in the waypoint-fixed coordinated frame that is invariant with respect to the desired approach direction χ_W. When a waypoint is reached, the next one is selected, the actual reference value W is changed and the rotation matrix is updated to transform position and orientation errors into the new waypoint coordinate frame.

As described above, the aircraft autopilots are designed to track desired airspeed (V_d), heading (χ_d) and flight path angle (γ_d).

Using the completed decoupled implementation of guidance laws, three independent Takagi-Sugeno Fuzzy Controller have been designed to constitute the FGS.

One FC generates the desired flight path angle (γ_d) for the autopilot using altitude error $e_H = (H_W - H)$:

$$\gamma_d = f_\gamma(e_H) \tag{6.11}$$

The second fuzzy controller computes desired aircraft velocity:

$$V_d = V_W + f_V(V_W - V) = V_W + f_V(e_V) \tag{6.12}$$

The third, and most complex FC is demanded to generate the desired heading angle (χ_d) using the position errors along the X and Y axis of the actual waypoint-frame $(e^w_{X_C}, e^w_{Y_C})$, and heading error e_χ. Fuzzy rule-set is designed at a fixed airspeed value, this fact can produce a lack of tracking performances when the desired waypoint crossing-speed V_W differs significantly from tune-up value. The solution to this

problem is achieved by introducing a speed-correlated scale coefficient to position errors.

Let:

$$\begin{pmatrix} e_X^w \\ e_Y^w \end{pmatrix} = Rot(\chi_W) \cdot \begin{pmatrix} e_X \\ e_Y \end{pmatrix} = \\ = Rot(\chi_W) \cdot \begin{pmatrix} X_W - X \\ Y_W - Y \end{pmatrix} \tag{6.13}$$

be position errors in the fixed waypoint coordinates frame, $(e_{X_C}^w, e_{Y_C}^w)$ is the velocity-compensated position errors defined by:

$$\begin{pmatrix} e_{X_C}^w \\ e_{Y_C}^w \end{pmatrix} = S(V_W, V^*) \cdot \begin{pmatrix} e_X^w \\ e_Y^w \end{pmatrix} \tag{6.14}$$

$$S(V_W, V^*) = \frac{V^*}{V_W} \tag{6.15}$$

Where V^* represents the airspeed value used during FGS membership rules design. In this way, position errors, used by the FGS to guide aircraft toward WP with desired approaching direction, result to be magnified when V_W (requested waypoint crossing-speed) is larger than V^* or reduced otherwise. The equation 6.15 may diverge if V_W goes to zero, however this is a non-operative condition because the requested waypoint crossing-speed should be defined accordingly to aircraft flight parameters. Definition of parameter S denotes a new degree of freedom in FGS tuning process and may also be defined using a non-linear function of (V^*, V_W) provided that $S(V_W, V^*) = 1$ when $V_W = V^*$. Afterwards the desired heading angle produced by FC is defined by:

$$\chi_d = \chi_W + f_\chi(e_{X_C}^w, e_{Y_C}^w, e_\chi) \tag{6.16}$$

Figure 6.1 shows the complete fuzzy guidance and control scheme.

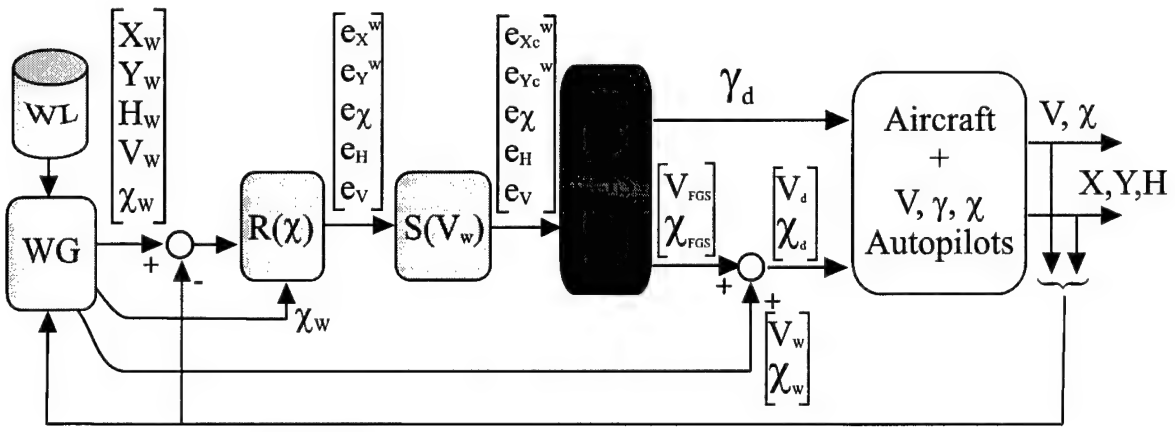


Figure 6.1: The First FGS SChematic

6.3.2 Takagi-Sugeno Fuzzy Controller Concepts

FGS is based on Takagi-Sugeno fuzzy systems[Takagi, 1985] model described by a blending of fuzzy IF-THEN rules:

$$\begin{aligned}
&IF\ x_1\ IS\ F_1^1\ AND\ x_2\ IS\ F_2^1\ AND\ \cdots \\
&\quad \cdots\ AND\ x_n\ IS\ F_n^1\ THEN\ y\ IS\ G_y^1 \\
&\\
&IF\ x_1\ IS\ F_1^2\ AND\ x_2\ IS\ F_2^2\ AND\ \cdots \\
&\quad \cdots\ AND\ x_n\ IS\ F_n^2\ THEN\ y\ IS\ G_y^2 \\
&\vdots \\
&\vdots \\
&IF\ x_1\ IS\ F_1^m\ AND\ x_2\ IS\ F_2^m\ AND\ \cdots \\
&\quad \cdots\ AND\ x_n\ IS\ F_n^m\ THEN\ y\ IS\ G_y^m
\end{aligned} \tag{6.17}$$

where $(x_1, \cdots, x_n)$ is the fuzzy controller (FC) input vector, F_j^i are fuzzy sets and y is the FC output. Using a weighted average defuzzifier layer each FC output should

be defined as follows:

$$y = \frac{\sum_{i=1}^{m} \mu_i(x) u_i}{\sum_{k=1}^{m} \mu_k(x)} \tag{6.18}$$

where $\mu_i(x)$ is the ith membership function of input x to ith fuzzy zones.

6.3.3 Fuzzy Guidance Laws Design

The fuzzy rules have been defined according to the desired approaching behavior and angular rates limitations of the aircraft. Fuzzy knowledge base was designed to generate flyable trajectories using the maximum linear and angular velocities and accelerations that are typical of a small propeller-engine aircraft. The BLUESNIPE research aircraft characteristicswere taken as a reference [Giulietti, F. *et al.*, 2000]. Anyhow, the laws can be easily adapted to the performances of any other aircraft and this implies varying the membership functions dimensions only. The FGS provides with different desired flight path and heading angle commands for different values of distance from the waypoint.

Deriving from the complete uncoupled aircraft model it is possible to describe each fuzzy controller separately from the others. The Altitude controller and the Velocity FC are less complex than Heading controller, they are both implemented using Takagi-Sugeno model; for the first one the only input is the altitude error $e_H = (H_W - H)$ and four fuzzy set are designed to map this input and four for the γ_d output:

- If e_H Is N_∞ Then γ_d Is P_{20}: for big negative errors.
- If e_H Is N_s Then γ_d Is P_2: for small negative errors.

- If e_H Is P_s Then γ_d Is N_2: for small positive errors.

- If e_H Is P_∞ Then γ_d Is N_{20}: for big positive errors.

The Velocity FC has similar complexity as the Altitude controller: 3 input fuzzy sets for e_V velocity error and 3 for the resulting ΔV_d output:

- If e_V Is N_∞ Then ΔV_d Is P_{10}: for negative errors.

- If e_V Is ZE Then ΔV_d Is ZE: for near to zero errors.

- If e_V Is P_∞ Then ΔV_d Is N_{10}: for positive errors.

In fact, we assumed the vehicle to be autopiloted in velocity, so this controller could be avoided but its usefulness lies in its nonlinear characteristic that acts as gain varying with error itself.

As stated before guidance in the horizontal $(X-Y)$ plane is more complex than guidance in the vertical $(X-H)$ plane. The horizontal plane fuzzy controller takes his input from scaled position errors $(e^w_{X_C}, e^w_{Y_C})$ and heading error (e_χ). $(e^w_{X_C})$ error is coded into five gaussian fuzzy sets:

- N_∞: for big negative errors.

- N_s: for small negative errors.

- ZE: for near to zero errors.

- P_s: for small positive errors.

- P_∞: for big positive errors.

three sets are also defined for $(e^w_{Y_C})$ (Y^w axes error): (N_s, ZE, P_s). Considering the Takagi-Sugeno output function for this fuzzy controller:

$$
\begin{aligned}
y &= \frac{\sum_{i=1}^{m} \mu_i(x) u_i}{\sum_{k=1}^{m} \mu_k(x)} = \\
&= \frac{1}{c(x)} \sum_{i=1}^{S} \sum_{j=1}^{K} \mu_i^{xy}(e^w_{X_C}, e^w_{Y_C}) \cdot \mu_{ij}^{\chi}(e_\chi) u_{ij} = \\
&= \frac{1}{c(x)} \sum_{i=1}^{S} \mu_i^{xy}(e^w_{X_C}, e^w_{Y_C}) \cdot \delta_{ij}^{\chi}(e_\chi)
\end{aligned}
\tag{6.19}
$$

where:

$$
\begin{aligned}
c(x) &= \sum_{k=1}^{m} \mu_k(x) \\
\mu_i^{xy}(e^w_{X_C}, e^w_{Y_C}) &= \mu_i^{x}(e^w_{X_C}) \cdot \mu_i^{y}(e^w_{Y_C})
\end{aligned}
\tag{6.20}
$$

S is the number of zones dividing the flight space and K is the number of subsets (dependent from e_χ) defined for each zone. Equation (6.19) can be simplified:

$$
\sum_{i=1}^{S} \frac{\mu_i^{xy}(e^w_{X_C}, e^w_{Y_C})}{c(x)} \cdot \delta_{ij}^{\chi}(e_\chi) = \sum_{i=1}^{S} \overline{\mu}_i^{xy}(e^w_{X_C}, e^w_{Y_C}) \cdot \delta_{ij}^{\chi}(e_\chi) \tag{6.21}
$$

Fixing $(e^w_{X_C}, e^w_{Y_C})$ in the middle of Pth zone, under the assumption that the contribution from the other zones is near to zero:

$$
\begin{aligned}
y\Big|_{\substack{e^w_{X_C^P} \\ e^w_{Y_C^P}}} &= \overline{\mu}_P^{xy}(e^w_{X_C^P}, e^w_{Y_C^P}) \cdot \delta_P^{\chi}(e_\chi) + \\
&+ \sum_{\substack{i=1 \\ i \neq P}}^{S} \overline{\mu}_i^{xy}(e^w_{X_C}, e^w_{Y_C}) \cdot \delta_{ij}^{\chi}(e_\chi) \cong \\
&\cong \overline{\mu}_P^{xy}(e^w_{X_C^P}, e^w_{Y_C^P}) \cdot \delta_P^{\chi}(e_\chi)
\end{aligned}
\tag{6.22}
$$

Equation (6.22) shows that the definition of fuzzy sets for (e_χ) error should be computed looking at each single set partitioning the flight space and then looking at the cumulative result. Under this assumption, seven fuzzy sets have been defined:

- N_b and P_b: used to keep output to $\pm\pi$ when big negative or positive errors.
- N_m and P_m: used to keep output to $\pm\frac{\pi}{2}$ when medium negative or positive errors.
- N_s and P_s: used to lead output to 0 when small negative or positive errors.
- ZE: used to keep output to 0 when near to zero errors.

These fuzzy sets have been designed considering a fixed aircraft velocity ($V^* = 25m/s$), figure 6.2 shows a contour plot of $u_i^{xy}(e^w_{X^P_C}, e^w_{Y^P_C})$ membership functions at fixed e_χ.

These fuzzy sets have been designed considering a fixed aircraft velocity ($V^* = 25m/s$), figure 6.2 shows a contour plot of $u_i^{xy}(e^w_{X^P_C}, e^w_{Y^P_C})$ membership functions at fixed e_χ.

6.4 First FGS simulation results

This section shows some simulation results for the first Fuzzy Guidance System. The presented simulation shows a non planar trajectory. First the aircraft is driven to waypoint W_1, then to align with W_2, then to W_3 that is 150 meters lower in altitude and very near on the (X, Y) plane and finally to W_4 that is at altitude 100 with a desired approach angle rotated by $\frac{\pi}{2}$ from previous waypoint. Figure 6.5 shows this trajectory. The required descent from W_2 to W_3 is too steep for the aircraft

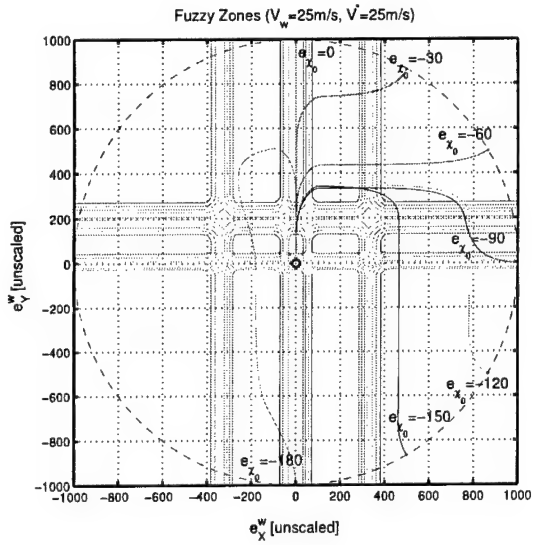


Figure 6.2: Contour plot of $u_i^{xy}(e^w_{X_C^P}, e^w_{Y_C^P})$ membership functions

capabilities, as decided in the design phase of fuzzy rule-set. As a matter of fact, when the aircraft reaches the X, Y coordinates of W3 its altitude is still too high, and it starts a turn to come back to the waypoint at the prescribed altitude. In fact, the aircraft begins a spiral or 8-shape descent, centered on the waypoint vertical axis, decreasing altitude with the descent rate limitation given by FGS, until the waypoint altitude is reached and then it proceeds to next waypoint. In this particular case, half turn is enough to reach the altitude of W_3, thus, when it reaches the desired altitude, it holds it and crosses successfully waypoint W_3 and, successively, waypoint W_4.

The above described manoeuvre has not been planned when setting waypoints; the FGS generates it only because the waypoints, as a matter of fact, describe a non flyable trajectory under the maximum accelerations design constraints.The design of fuzzy sets requires a good knowledge of the Takagi-Sugeno fuzzy systems. Starting

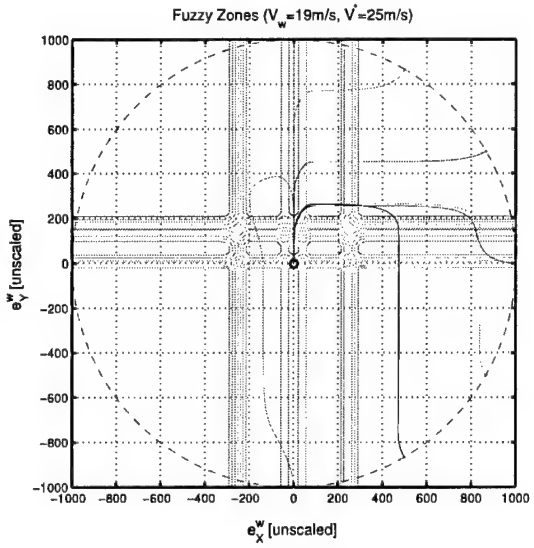


Figure 6.3: Contour plot of $u_i^{xy}(e^w_{X^P_C}, e^w_{Y^P_C})$ Membership Functions

from desired flight performances fuzzy sets should be defined in accordance to the aircraft flight constraints. Relationships between requirements and fuzzy sets are not trivial, thus a specific graphic tools has been developed under Matlab in order to aid the fuzzy designer in the most accurate analysis of the fuzzy controller behavior.The proposed tool is capable to load files from Matlab Fuzzy toolbox and allows to plot the output of the fuzzy system in different flight conditions.

6.5 Problems with the first FGS

The first FGS is capable of achieving the first four requirements of the guidance system but it has some limitations:

1. loss of accuracy under some conditions.

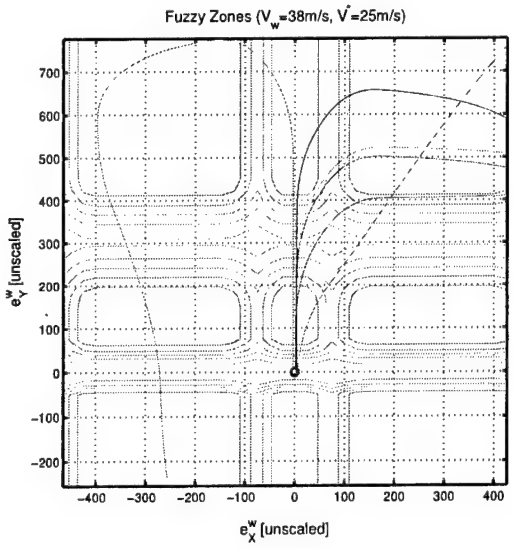


Figure 6.4: Contour plot of $u_i^{xy}(e^w_{X_C^P}, e^w_{Y_C^P})$ membership functions

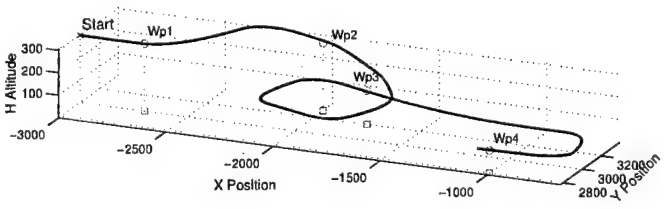


Figure 6.5: Simulation of 4 waypoints trajectory

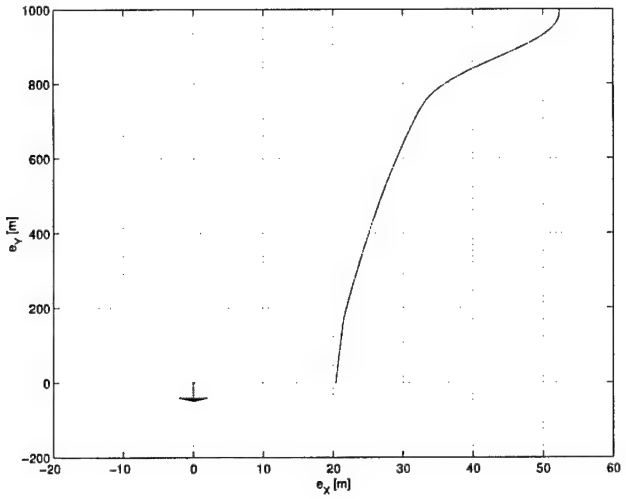


Figure 6.6: Example of Miss Distance Error

2. Presence of some singularities that raise incorrect control signals

3. High number of fuzzy rules that make behavior analysis difficult.

6.5.1 Accuracy

There are unapcetable errors in the miss distance for some initial state configurations as shown in (fig.6.6) where the miss distance is 20 meters.

The aircraft initial position is $x = 50, y = 1000$ meters, in the waypoint reference system, with no heading error. The problem arises because of the rules set fpr the zone XisZE, lwhose membership function has an extension of 40 meters with the origin in the middle. Such fuzzy rules were designed assuming that the vehicle crosses their frontiers with a given heading, because of provenience from neighbor plane areas, that is adjacent fuzzy areas. If the heading angle value is lower than the designed value,

the waypoint is not reached.

6.5.2 Singularities

To explain the second limitation, consider two points having the same ordinate, in the waypoint reference system, for instance: $x1 = -500$[m] and $x2 = 500$[m] with $y = 1000$[m]. Consider now the output of controller $FLC_\chi(e_X, e_Y, e_\chi)$ computed along the line connecting the two points for an angle error $e_\chi = \pm 180^o$ coinsiding with the singularity:

$$[e_X, e_Y]^T = [x1, y]^T \lambda + [x2, y]^T (1 - \lambda) \ , \ \lambda \in [0, 1]$$

The output is shown in Figure (6.7: from the continuity of $FLC\chi$ there exists a value of e_X for which the system output is in fact $\pm 180^o$. This implies that an aircraft in that position with that heading continues its route without turning toward the waypoint. As a result of this the system may show limit cycles.

6.5.3 Fuzzy rules

Redundancy of fuzzy rules in the single zones is another aspect that needs to be taken into account. For each zone, rule impose a specific exit direction. This value determines in the graph an intersection point with the unit slope curve, where FLC_χ must have negative slope. For instance in Figure (6.5.3) the fuzzy controller output is shown for X*is*ZE *and* Y*is*ZE. The controller output function determines the appropriate speed, necessary to reach the waypoint with the required heading.

6.6 The second FGS structure

The limitations outlined above required a modification of the overall controller.

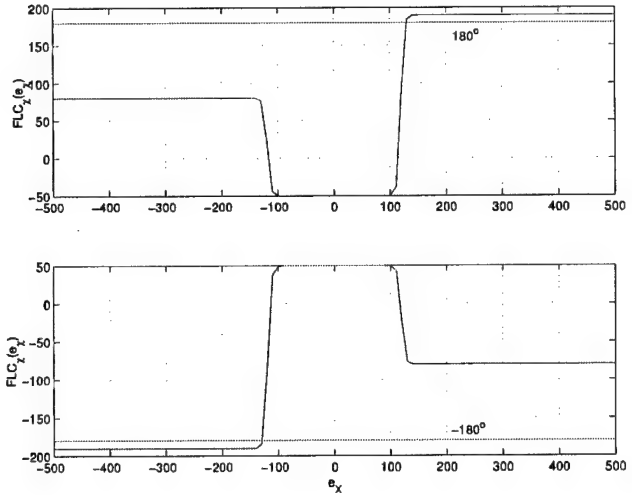
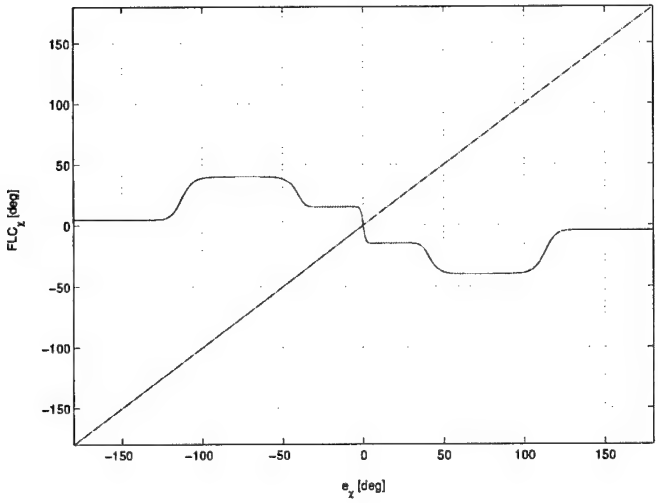


Figure 6.7: Output of Controller $FLC_\chi(e_X, e_Y, e_\chi)$



- From 1) the central zone is eliminated and substituted by an interpolation of zones XisN e XisP.
- Since the modification may introduce trajectories possessing a limit cycle between zones XisN e XisP, the error measure was modified so that the singularity point was avoided.
- From 3) and the above modification, the number of rules was decreased.

Starting from the previous FGS scheme, it will be shown how the new solution changed the heading controller. In the remaining of this document, the variables that refer to the waypoint will have a W suffix, those for the vehicle/aircraft will have the A suffix; later the moving waypoint or target variables will have the T suffix.

The first FGS logic can be summarized as follows:

$$\begin{cases} V_d = & V_W + f_V(e_V) \\ \gamma_d = & f_\gamma(e_H) \\ \chi_d = & \chi_W + f_\chi(e^w_{X_C}, e^w_{Y_C}, e_\chi) \end{cases} \tag{6.23}$$

where

$$\begin{cases} e_V & = V_A - V_W \\ e_H & = H_A - H_W \\ \left[e^w_{X_C}, e^w_{Y_C}\right]^T & = S(V_W, V^*) \cdot R^T(\chi_W) \cdot [X_A - X_W, Y_A - Y_W]^T \\ e_\chi & = \chi_A - \chi_W \end{cases} \tag{6.24}$$

with

$V_A =$	modulus of aircraft velocity
$V_W =$	modulus of desired crossing velocity of current waypoint
$H_A =$	aircraft altitude
$H_W =$	waypoint altitude
X_A, Y_A	aircraft position in the XY-plane
X_W, Y_W	waypoint position in the XY-plane
$\chi_A =$	aircraft heading
$\chi_W =$	desired crossing heading of current waypoint

The new FGS structure, for the heading controller, varies significantly . The new control logic is the following:

$$\begin{cases} V_d = & V_W + f_V(e_V) \\ \gamma_d = & f_\gamma(e_H) \\ \chi_d = & \widehat{\chi}_W + FLC_\chi(\widehat{e}_\chi, V_A) \end{cases} \tag{6.25}$$

where $\widehat{\chi}_W$ e $\widehat{e}_\chi$ are a new definition for the crossing heading and the route errorrespectively:

$$\begin{cases} \widehat{\chi}_W = & \chi_W + \delta\chi_W(e^w_{X_C}, e^w_{Y_C}) \\ \widehat{e}_\chi = & \chi_A - \widehat{\chi}_W \end{cases} \tag{6.26}$$

where $\delta\chi_W$ is the fuzzy controller that generates the desired route heading in every point of XY-plane. This computation is performed in waypoint-fixed frame coordinates. This new reference $\widehat{\chi}_W$ is used by a second fuzzy controller to compute the new heading error $\widehat{e}_\chi$ to be used as reference for aircraft heading autopilot. The heading controller now has a pipeline of two stages composed of two fuzzy systems much simpler than the unique one of the first FGS.

The resulting heading controller is depicted in figure 6.8.

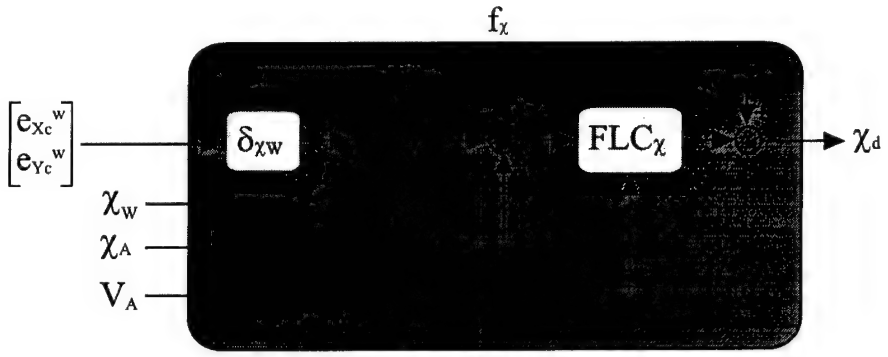


Figure 6.8: The second FGS Scheme

6.6.1 First stage: desired route

Yhe first stage generates the heading value to be reached within each zone in the waypoint fixed reference. The design of this stage depends directly from desired vehicle trajectories and must take into account maximum aircraft performance. Success in designing the first stage fuzzy system results in smooth maneuvers and limited control signals.

To correctly approach the problem of defining desired trajectories in every point of the horizontal plane, the trajectories are defined as a convex interpolation of desired heading in a finite number of partitions of the whole plane. The T-S fuzzy system succesfully accomplishes this task; because of the definition of position error in Cartesian coordinates and because of the nature of fuzzy membership functions, the areas have rectangular bounds.

To avoid the steep transitions in the desired route especially in the vicinity of waypoint axis origin, which characterized the fist FGS, the problem has been subdivided in two subproblems: guidance in the upper half plane ($e_Y > 0$) and in the lower half

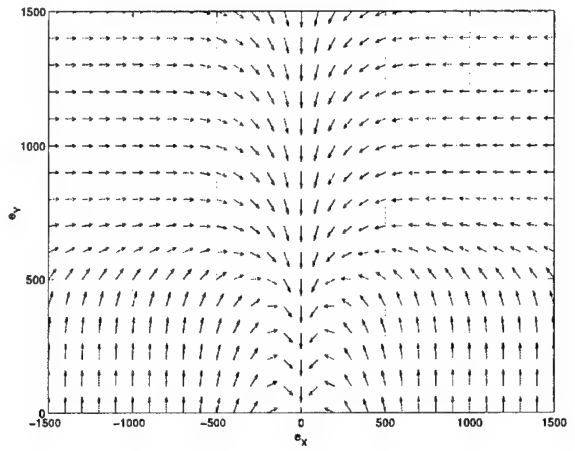


Figure 6.9: Desired routes in the upper plane ($e_Y > 0$)

plane ($e_Y < 0$).

The graphics of the interpolated desired routes in the upper and lower planes are shown in figures (6.9) and (6.10).

To take into account the singularities in heading representation, namely the fact that heading measure has a discontinuous representation with a value of $\pm 180^o$, the output of the two fuzzy controllers must be corrected of $\pm 360^o$ depending on aircraft heading. The idea that lies beneath this correction is to generate a reference so that an heading error exceeding $\pm 180^o$ is never achieved. This implies that $|\widehat{e}_\chi| = |\chi_A - \widehat{\chi}_W| = |\chi_A - X_W - \delta\chi_W| < 180^o$. As an example, consider the following:

Suppose $e_X = -1000$,$e_Y = 1000$[m] with $e_\chi = \chi_A - \chi_W = -170^o$, that is on the upper plane and away from the waypoint. The direction computed by the first stage is: $\delta\chi_W = 90^o$, thus a counterclockwise turn would be imposed to the vehicle:

$\widehat{e}_\chi = \chi_A - \widehat{\chi}_W = \chi_A - \chi_W - \delta_\chi W = e_\chi - \delta_\chi W = -170 - 90 = -260$

To avoid this problem, the $\delta_\chi W$ value is corrected by a factor $c = -360^o$. This

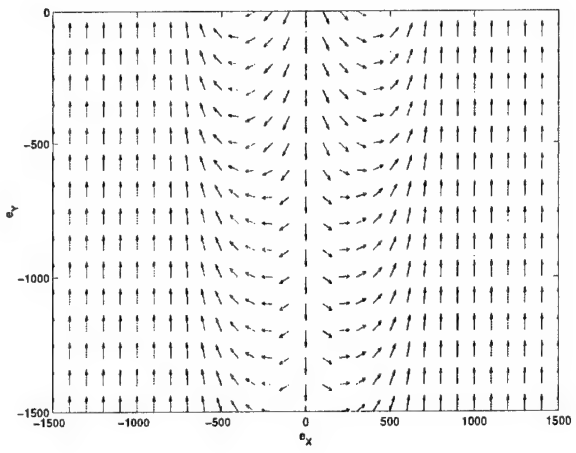


Figure 6.10: Desired routes in the lower plane ($e_Y < 0$)

approach is like rotating the reference χ_W depending on the vehicle heading so that the point $e_\chi = \pm 180^o$. The resulting $|\widehat{e}_\chi| = |-260-(-360)| = |100| < 180$ is inside desired limits.

In order of mixing smoothly the guidance in the upper and lower plane, the corrected outputs of the two modules are interpolated with a weigthing function μ_Y that depends on e_Y only:

$$\delta\chi_W = \mu_Y(e_Y)\cdot(FLC_{sup}(e_X,e_Y)+c_{sup})+(1-\mu_Y(e_Y))\cdot(FLC_{inf}(e_X,e_Y)+c_{inf}) \quad (6.27)$$

where $c_{inf} = \pm 2k\pi$ e $c_{sup} = \pm 2h\pi$ e $k,h = \pm 0, \pm 1, ...$ Therefore the reference generated by the first stage is:

$$\widehat{\chi}_W = \chi_W + \mu_Y(e_Y)\cdot(FLC_{sup}(e_X,e_Y)+c_{sup})+(1-\mu_Y(e_Y))\cdot(FLC_{inf}(e_X,e_Y)+c_{inf}) \quad (6.28)$$

The first stage is in fact composed of the two fuzzy controllers FLC_{sup} and FLC_{inf} and their mixer.

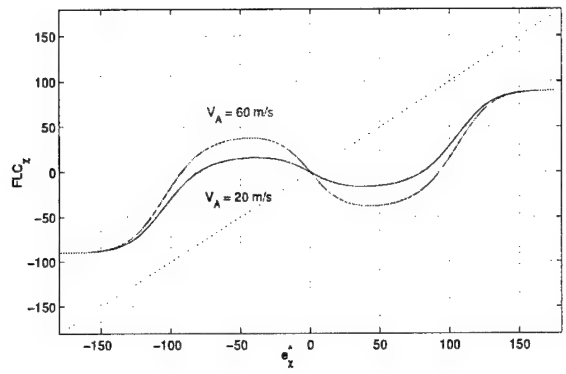


Figure 6.11: Function $FLC_\chi(\widehat{e}_\chi, V_A)$

6.6.2 Second stage: heading error

The second stage generates the heading angle based on the error between the aircraft heading χ_A and the reference value generated by the first stage $\widehat{\chi}_W$. The objective here is to make $\widehat{e}_\chi = \chi_A - \widehat{\chi}_W \in [-180^o, 180^o]$ equal to zero. The second stage output is:

$$\chi_{rif} = \widehat{\chi}_W + FLC_\chi(\widehat{e}_\chi, V_A)$$

To prove the correctness of this stage logic, suppose $\chi_A > \widehat{\chi}_W$, then to decrease the heading tracking error, $\dot{\chi}_A < 0$, that is $\chi_d < \chi_A$, must hold because of the linear dynamics expressed by equation (6.10); then $FLC_\chi(\widehat{e}_\chi, V_A) < \widehat{e}_\chi$. Otherwise if $\chi_A < \widehat{\chi}_W$, $\dot{\chi}_A > 0$ must hold and $FLC_\chi(\widehat{e}_\chi, V_A) > \widehat{e}_\chi$. Then $FLC_\chi(\widehat{e}_\chi, V_A) \in [-180^o, 180^o]$ and the problem can be solved as convex combination of constant coefficients in the range $[-180^o, 180^o]$.

The dependency of $FLC_\chi(\widehat{e}_\chi, V_A)$ from the vehicle's velocity, allows to limit the increase of radius of curvature as velocity increases. The behavior of $FLC_\chi(\widehat{e}_\chi, V_A)$ for two different speed is shown in Figure 6.11.

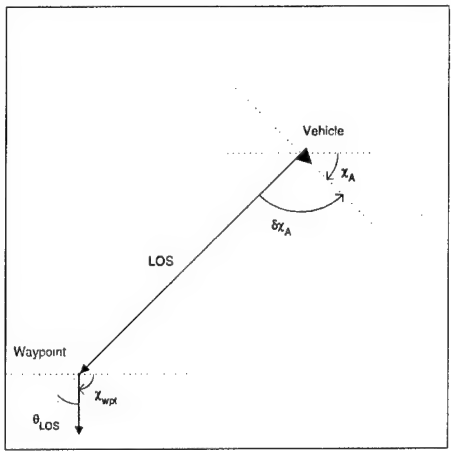


Figure 6.12: LD-*guidance* geometry

6.6.3 Long and short Distance Guidance

When the vehicle is away from the waypoint, a different guidance is used, in order to optimize the shape of the trajectories. This guidance is called $Long - Distance\ Guidance$ (LD). The definition of (LD) guidance is set by a distance from the waypoint, which is a design parameter and set, in our simulations, equal to a circle of radius $R = 2500$[m]. A blending law that depends on the range R computed as $||R_{AW}|| = \sqrt{e_X^2 + e_Y^2}$ selects either LD-*guidance* or SD-*guidance* :

$$\chi_d = \chi_W + \mu_R(R_{AW}) \cdot F_{LD}(e_X, e_Y, \chi_A, \chi_W) + (1 - \mu_R(R_{AW})) \cdot F_{LD}(e_X, e_Y, \chi_A, \chi_W) \tag{6.29}$$

LD-*guidance* is defined using the quantities in Figure 6.12. The guidance system makes the aircraft reach the LOS direction, zeroing the error $\delta\chi_W = \chi_A - (\chi_W + \theta_{LOS})$.

The computation of LOS angle is performed projecting the position error from

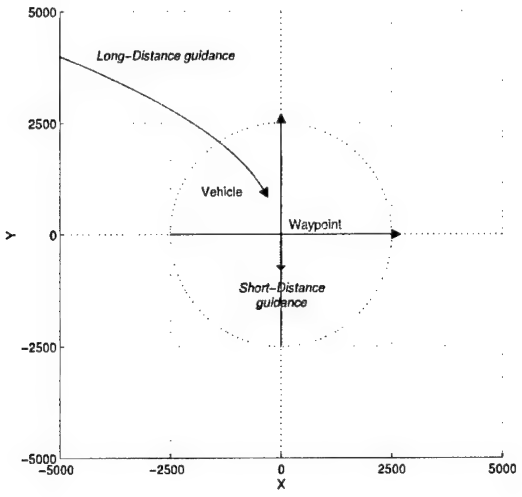


Figure 6.13: Example of long-short distance mixing

the Earth axes to a reference at an angle χ_W .Therefore θ_{LOS} is given by:

$$\theta_{LOS} = \texttt{atan2}(e_X, -e_Y) \tag{6.30}$$

The heading error entering the LD-*guidnace* fuzzy controller is:

$$\delta\chi_W = \chi_A - (\chi_W - \theta_{LOS}) \tag{6.31}$$

and for $\mu_R(R) = 1$ we have:

$$\chi_d = \chi_W + \theta_{LOS} + FLC_{LD}(\delta_{\chi A}) \tag{6.32}$$

Equation (6.32) is now in the appropriate form for blending with SD-*guidance.*

The resulting FGS is depicted in Figure 6.14 and the complete control loop with the second FGS is depicted in Figure 6.6.3.

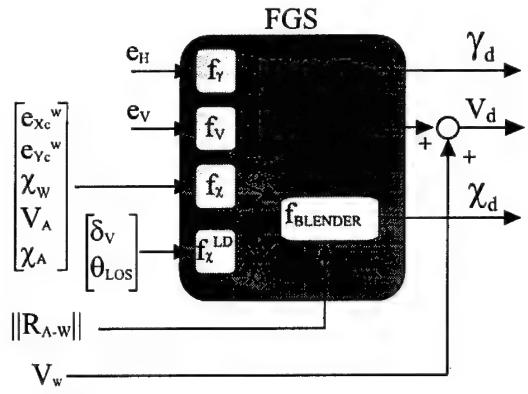
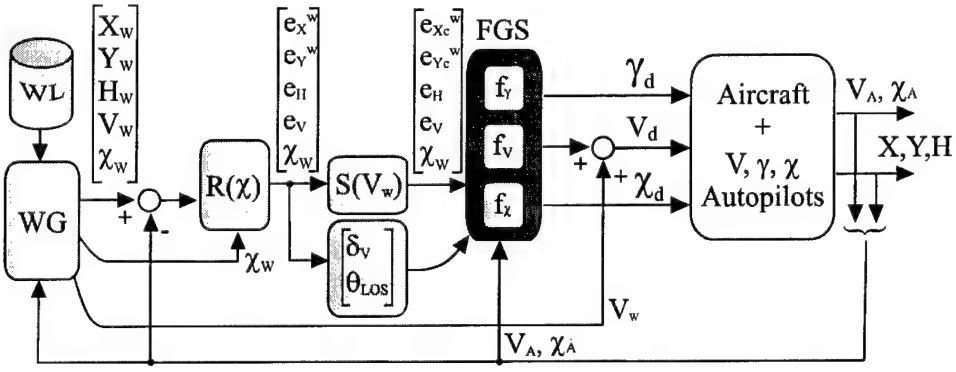


Figure 6.14: The second FGS scheme



6.7 Management of moving Waypoints

The position error dynamics in the earth reference system, in the case of static waypoint, depends on the aircraft velocity only:

$$[\dot{e}_X, \dot{e}_Y]^T = \underline{V}_A - \underline{V}_T \tag{6.33}$$

where X_A, Y_A and $X_T, Y_T\}$ are the position of the vehicle and the waypoint along the X and Y axis of earth-frame, $\bar{E}_{XY}$ is the vector position error and $\dot{\bar{E}}_{XY}$ is the rate of change of position error.

When the waypoint to be reached is moving (a model of a target, for instance), the second of (6.33) no longer holds, and $\bar{E}_{XY}$ depends on the relative velocity. The tracking performance

degrades rapidly as a function of the waypoint speed $\bar{V}_T$ and the system cannot guarantee the waypoint interception even in easy cases. Let us consider, for instance, a vehicle aligned with the desired crossing direction, that is $\chi_A = \chi_W$, and placed in an interception triangle with the waypoint which is moving in uniform linear motion at a given speed $\bar{V}_T$. Figure 6.15 shows this situation. As a consequence of the intercept condition, the vector $\bar{V}_A - \bar{V}_T$ results aligned with the line of sight (LOS). The FGS is able to maintain such condition only if the position error along waypoint frame X axis, is zero. In this case, $FLC_\chi(\chi) = 0$ and the vehicle heading remains equal to the crossing headingχ_W. If $\bar{V}_T \neq 0$, the condition $e_\chi = 0$ will never be verified with an intercept triangle, and proposed system does not guarantee interception with a zeroposition error.

By means of appropriate modifications to the error measures, the proposed FGS will successfully achieve interception even in the uniform straight motion case. Let us

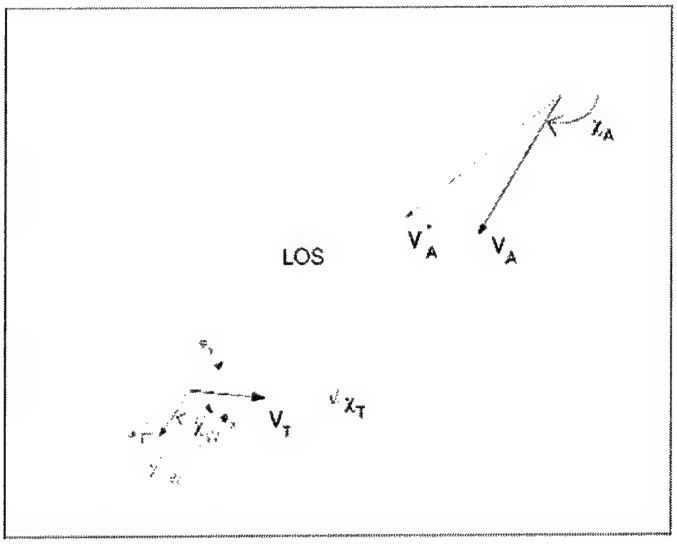


Figure 6.15: Intercept Conditions with moving Waypoint

define $\bar{P}_A(t)$ as the current vehicle position. Consider a *virtual* vehicle with position $\bar{P}_A^*(t)$ such that $\bar{P}_A(t_0) = \bar{P}_A^*(t_0)$ at instantt_0. Define the virtual vehicle velocity as $\bar{V}_A^*(t) = \bar{V}_A(t) - \bar{V}_A^*(t)$

The the virtual vehicle position at time t is:

$$\bar{P}_A^*(t) = \bar{P}_A^*(t_0) + \int_{t_0}^{t} \bar{V}_A^*(\tau) d\tau \tag{6.34}$$

that can be written as:

$$\bar{P}_A^*(t) = \bar{P}_A(t) - \int_{t_0}^{t} \bar{V}_T(\tau) d\tau \tag{6.35}$$

Adding and substracting $\bar{P}_T(t_0)$ to the right hand side yields

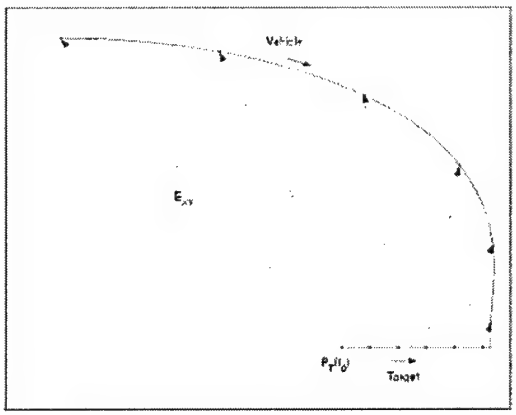


Figure 6.16: Intercept Example, Target with uniform Motion

$$\begin{aligned} \bar{P}_A^*(t) &= \bar{P}_A(t) - \int_{t_0}^{t} \bar{V}_T(\tau)d\tau + \bar{P}_T(t_0) - \bar{P}_T(t_0) = \\ &= \bar{P}_A(t) - \bar{P}_T(t) + \bar{P}_T(t_0) = \bar{E}_{XY} + \bar{P}_T(t_0) \end{aligned} \tag{6.36}$$

since $\bar{P}_T(t) = \bar{P}_T(t_0) + \int_{t_0}^{t} \bar{V}_T(\tau)d\tau$.Bringing $\bar{P}_T(t_0)$ to the left hand side yields:

$$\bar{P}_A^*(t) - \bar{P}_T(t_0) = \bar{P}_A(t) - \bar{P}_T(t) = \bar{E}_{XY} \tag{6.37}$$

Equation (6.37) indicates that when the real vehicle reaches the target ($\bar{E}_{XY} = 0$), the virtual vehicle is in the initial target position $\bar{P}_T(t_0)$. The position error $\bar{E}_{XY}$ can also be seen as the error between the virtual vehicle and the initial target (waypoint) position. Figure 6.16 shows the position error vector during an interception. Differentiating equation (6.37), and recalling that $\dot{\bar{P}}_T(t_0) = 0$ yields:

$$\dot{\bar{P}}_A^*(t) = \bar{V}_A^* = \bar{V}_A(t) - \bar{V}_T(t) = \dot{\bar{E}}_{XY} \tag{6.38}$$

that is the rate of change of position error coincides with the virtual vehicle velocity $\dot{\bar{P}}_A^*(t)$. When the waypoint is moving in uniform straight motion, equation (6.33) is replaced by:

$$\begin{cases} \overline{E}_{xy} = [X_A - X_T, Y_A - Y_T]^T = [X_A^* - X_T(t_0), Y_A^* - Y_T(t_0)]^T \\ \dot{\overline{E}}_{xy} = \bar{V}_A^* \end{cases} \tag{6.39}$$

The position error in the waypoint coordinate frame $\bar{e}_{XY}$ results:

$$\bar{e}_{XY} = R^T\left(\chi_W + \frac{\pi}{2}\right) \cdot \bar{E}_{XY} \tag{6.40}$$

If χ_W is constant, then $\dot{R}^T\left(\chi_W + \frac{\pi}{2}\right) = 0$, and the position error dynamics become

$$\dot{\bar{e}}_{XY} = R^T\left(\chi_W + \frac{\pi}{2}\right) \cdot \dot{\bar{E}}_{XY} \tag{6.41}$$

from the second of (6.33), the position error rate of change between the real vehicle and target depends on the virtual vehicle velocity only:

$$\dot{\bar{e}}_{XY} = R^T\left(\chi_W + \frac{\pi}{2}\right) \cdot \bar{V}_A^* \tag{6.42}$$

Since when the waypoint is fixed the virtual vehicle coincides with the real one, the FGS can always produce the desired heading χ_d for the real aircraft by means of the virtual vehicle position error respect to $\bar{P}_T(t_0)$and the virtual vehicle heading χ_A^* computed from:

$$\chi_A^* = \angle\left(\bar{V}_A - \bar{V}_T\right) \tag{6.43}$$

that is:

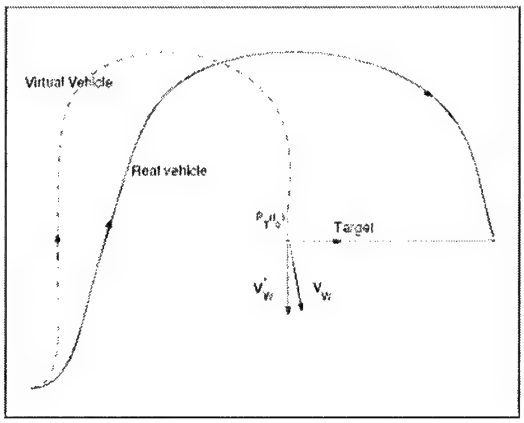


Figure 6.17: Real and Virtual Aircraft, Intercept Example

$$\bar{V}_A^* = V_A^* \begin{bmatrix} \cos \chi_A^* \\ \sin \chi_A^* \end{bmatrix} \tag{6.44}$$

Figure 6.17 shows real and virtual aircraft in an intercept example.

To satisfy the original guidance requirements, the following two conditions must be satisfied:

- The module of virtual vehicle's velocity must be kept to the value $\bar{V}_W$.
- The value of real vehicle heading angle, when crossing the waypoint, must be $\chi_A = \chi_W$.

The first condition yields:

$$\left\| \bar{V}_A^* \right\| = \left| \bar{V}_A - \bar{V}_A^* \right| = V_W \tag{6.45}$$

Squaring both sides and replacing the expression of the vehicle and target velocities:

$$V_W^2 = V_A^2 + V_T^2 - 2V_A V_T \left(\cos\chi_A \cos\chi_T + \sin\chi_A \sin\chi_T\right) \tag{6.46}$$

Looking for the solution for which $V_A > V_T$ we have

$$\begin{aligned} V_A = &+V_T \cos\left(\chi_A - \chi_T\right) + \\ &+\sqrt{V_W^2 - V_T^2 \sin^2\left(\chi_A - \chi_T\right)} = \widehat{V}_d\left(\chi_A\right) \end{aligned} \tag{6.47}$$

The second condition is satisfied with the introduction of a different desired heading angle at the waypoint. When the real vehicle reaches the waypoint with the correct velocity $V_A = \widehat{V}_d\left(\chi_A\right)$ and the correct heading angle $\chi_A = \chi_W$, then the virtual vehicle velocity is:

$$\bar{V}_A^* == \widehat{V}_d\left(\chi_W\right) \begin{bmatrix} \cos\chi_A^* \\ \sin\chi_A^* \end{bmatrix} - V_T \begin{bmatrix} \cos\chi_A^* \\ \sin\chi_A^* \end{bmatrix} \tag{6.48}$$

>From (??) and (6.48), the virtual vehicle heading angle, when crossing the initial waypoint/target position $\bar{P}_T\left(t_0\right)$ is:

$$\chi_W^* = \angle\left\{ \widehat{V}_d(\chi_W) \begin{bmatrix} \cos\chi_A^* \\ \sin\chi_A^* \end{bmatrix} - V_T \begin{bmatrix} \cos\chi_A^* \\ \sin\chi_A^* \end{bmatrix} \right\} \tag{6.49}$$

It can be shown that $\chi_W^* = \chi_W$ when $V_T = 0$. Therefore the desired heading angle for the virtual vehicle χ_W^* depends from the target velocity V_T; thus the error position in the waypoint frame is:

$$\bar{e}_{XY}^* = R^T\left(\chi_W^* + \frac{\pi}{2}\right) \cdot \bar{E}_{XY} \tag{6.50}$$

If the waypoint is moving in uniform straight motion, under the condition $\dot{\chi}_W = 0$, then $\dot{\chi}^*_W = 0$ and :

$$\dot{\bar{e}}^*_{XY} = R^T\left(\chi^*_W + \frac{\pi}{2}\right) \cdot \dot{\bar{E}}_{XY} = R^T\left(\chi^*_W + \frac{\pi}{2}\right) \cdot \bar{V}^*_A \tag{6.51}$$

The above results require modifications on both LD-guidance and SD-guidance. In SD-guidance the velocity ($\bar{V}_d$), and flight path angle (γ_d) references for the autopilots are generated according to the following:

$$\begin{cases} V_d = \widehat{V}_d\left(\chi_A\right) + f_V(e_V) \\ \gamma_d = f_H(e_H) \end{cases} \tag{6.52}$$

where $e_V = V_A - \widehat{V}_d\left(\chi_A\right)$ is the error between the real vehicle velocity and the desired velocity given in (6.47).

For the heading reference (χ_d). The first stage geneates the desired heading for the virtual vehicle

$$\chi^*_W = \chi^*_W + \delta_{\chi W}\left({}^w e^*_{X_C}, {}^w e^*_{Y_C}\right) \tag{6.53}$$

where $\left({}^w e^*_{X_C}, {}^w e^*_{Y_C}\right)$ are the velocity-compensated position errors along the waypoint frame X and Y axes:

$$\begin{aligned} &\left[{}^w e^*_{X_C}, {}^w e^*_{Y_C}\right]^T = S\left(V_W\right) \cdot \bar{e}^*_{XY} = \\ &= S\left(V_W\right) \cdot R^T\left(\chi^*_W + \tfrac{\pi}{2}\right) \cdot \bar{E}_{XY} \end{aligned} \tag{6.54}$$

The desired heading for the real aircraft , with $\varepsilon_A = \chi^*_A - \chi_A$:

$$\widehat{\chi}^*_W = \chi^*_W + \varepsilon_A \tag{6.55}$$

The reference heading χ_d for the autopilot becomes:

$$\chi_d = \widehat{\chi}_W + FLC_{\chi_W}\left(\widehat{e}_\chi, V_A\right) \tag{6.56}$$

$$\widehat{e}_\chi = \chi_A - \widehat{\chi}^*_W.$$

The LD-guidance has a different generation of desidered heading, equation (6.53) is replaced by:

$$\widehat{\chi}^*_W = \chi^*_W + \theta^*_{LOS} + FLC_{LD}\left(\delta_{\chi^*_A}\right) \tag{6.57}$$

with $\theta^*_{LOS} = a\tan 2\left({}^w e^*_{X_C}, {}^w e^*_{Y_C}\right)$ LOS angle with respect to χ^*_W, and $\delta_{\chi^*_A} = \chi^*_A - (\chi^*_W + \theta^*_{LOS})$. Equations (6.55), and (6.56) still hold.

6.7.1 Simulation Results

This section contains some simulation results with moving waypoints, results for fixed waypoints can be found in the interim report. In all simulations the target velocity is 10m/s, and the desired crossing relative velocity is 30m/s. The target is always reached within a good approximation in heading and intercepts distance. Figures 6.18 and 6.19 show two different simulation scenarios.

Figure 6.20 shows the case of two consecutive interceptions of a moving target.

The application of the proposed guidance to the case of targets moving with constant angular velocity is shown in the next two figures. In Figure 6.21 intercept is achieved, whereas in Figure 6.22 the miss distance is different from zero. This latter result may be satisfactory depending on the tolerance required.

Figure 19:

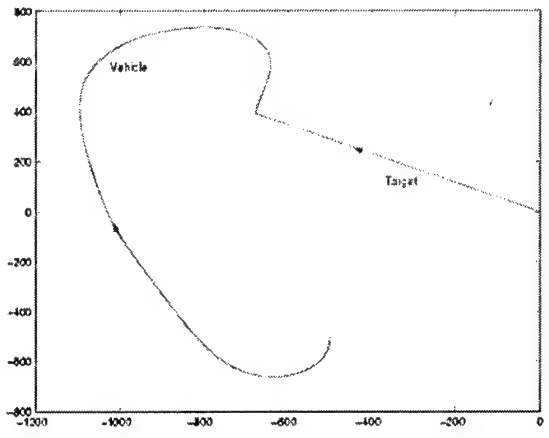


Figure 6.18: Simulation 1 with moving Target

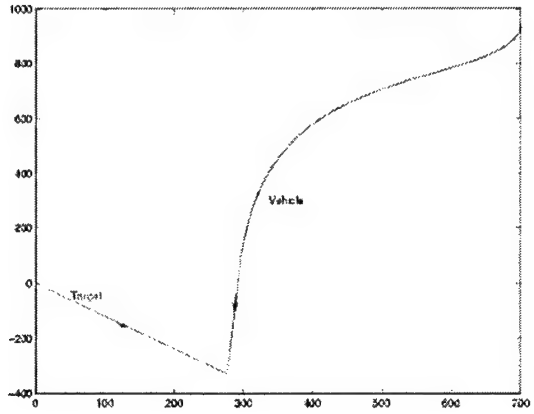


Figure 6.19: Simulation 2 with moving Target

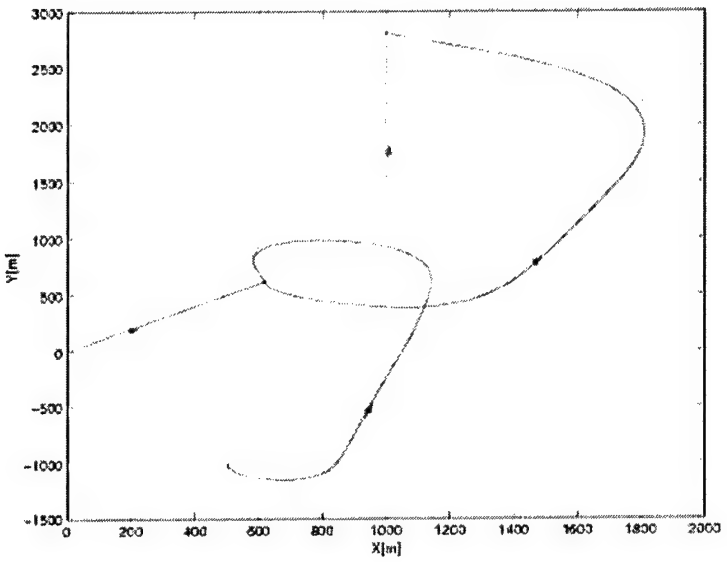


Figure 6.20: Two-Target Interceptions (Target in red)

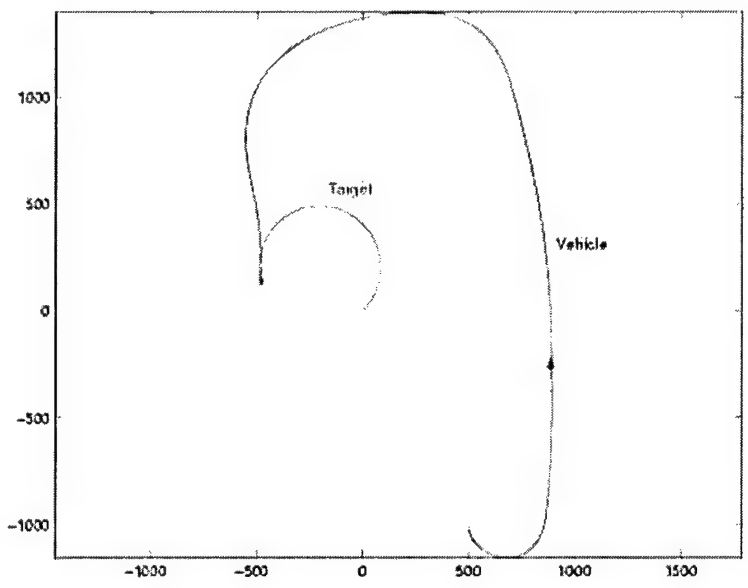


Figure 6.21: Target with nonzero Angular Velocity

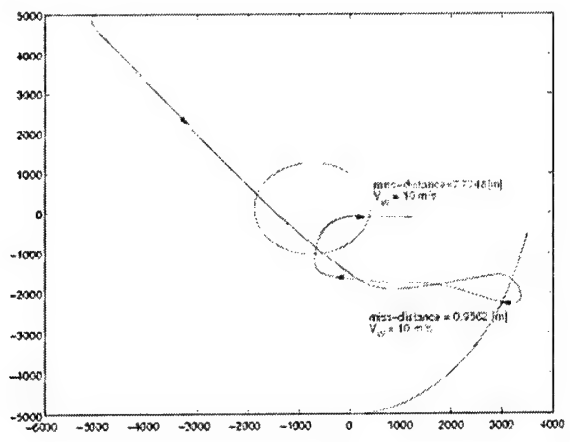


Figure 6.22: Intercept Scenario: Targets with an angular velocity equal to $\dot{\chi}_T = 2[deg/sec]$.

The las simulation shows the capability of the system to handle multiple vehicles and multiple targets.

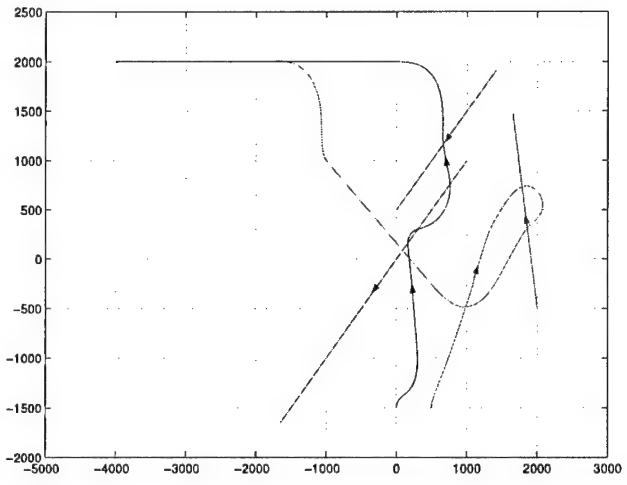


Figure 6.23: Multiple Vehiclas and moving Targets

Bibliography

[Anderson and Robbins, 1998] M. R. Anderson and A. C. Robbins. "Formation Flight as a Cooperative Game". In *Proceedings of Aiaa Guidance, Navigation and Control Conference*, Philadelphia, PA, August 1998.

[Blake, W. and Multhopp, D., 1998] Blake, W. and Multhopp, D. "Design, Performance and Modeling Considerations for Close Formation Flight". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, Philadalphia, PA, August 1998.

[Blake, W., 2000] Blake, W. "An Aerodynamic Model for Simulation of Close Formation Flight". In *Proceedings of AIAA Modeling and Simulation Technologies Conference*, Denver, CO, August 2000.

[Bloy, A. W. and Jouma'a, M., 1995] Bloy, A. W. and Jouma'a, M. "Lateral and Directional Stability Control in Air-to-Air Refuelling". *ImechE, Part G Journal of Aerospace Engineering*, Vol. 209 : pp. 299–305, 1995.

[Bloy, A. W. *et al.*, 1993] Bloy, A. W., West, M. G., Lea, K. A., and Jouma'a, M. "Lateral Aerodynamic Interference Between Tanker and Receiver in Air-to-Air Refuelling". *Journal of Aircraft*, Vol. 30 : pp. 705–710, 1993.

[Boyd, 1994] S. Boyd. *"Linear Matrix Inequalities in System and Control Theory"*. SIAM, 1994.

[Buzogany, L. E. *et al.*, 1993] Buzogany, L. E., Pachter, M., and D'Azzo, J. J. "Automated Control of Aircraft in Formation Flight". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, Monterey, CA, August 1993.

[Capetta, R. *et al.*, 2001] Capetta, R., Giulietti, F., and Innocenti, M. "WakeCAD: Aerodynamic Interference Calculation Toolbox for Design, Simulation and Control of UAVs". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, Montreal, Quebec, Canada, August 2001.

[Chichka, D. and Speyer, J., 1996] Chichka, D. and Speyer, J. "Decentralized Controllers for Unmanned Aerial Vehicle Formation Flight". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, San Diego, CA, July 1996.

[D. F. Chichka *et al.*, 1999] D. F. Chichka, J. L. Speyer, and C. G. Park. "Peak-Seeking Control with Application to Formation Flight". In *Proceedings of IEEE Conference on Decision and Control*, December 1999.

[F. Giulietti *et al.*, 2000] F. Giulietti, L. Pollini, and M. Innocenti. "Autonomous Formation Flight". *IEEE Control Systems Magazine*, 20(6):34–44, December 2000.

[Gingras, D. and Player, J., 2001] Gingras, D. and Player, J. "Static and Dynamic Wind Tunnel testing of Air Vehicles in Close Proximity". In *Proceedings of AIAA Atmospheric Flight Mechanics Conference*, Montreal, Quebec, Canada, August 2001.

[Gingras, D., 1999] Gingras, D. "Experimental Investigation of a Multi-Aircraft Formation". In *Proceedings of AIAA Applied Aerodynamics Conference*, 1999.

[Giulietti, F. *et al.*, 2000] Giulietti, F., Pollini, L., and Innocenti, M. "SNIPE: Development of an Unmanned Aerial Vehicle at DSEA". In *Proceedings of 15th Bristol International Conference on UAVs*, Bristol, UK, April 2000.

[G.Mancino *et al.*, 1999] M. Innocenti G.Mancino, M. Garofoli, and M. Napolitano. "Preliminary Analysis of Formation Flight Management". In *Proceedings of Aiaa Guidance, Navigation and Control Conference*, Portland, OR, August 1999.

[Godbole, D. N. *et al.*, 2000] Godbole, D. N., Lygeros, J., Singh, E., Deshpande, A., and Lindsey, A. E. "Communication Protocols for a Fault-Tolerant Automated Highway System". *IEEE Transaction on Control Systems Technology*, Vol. 8 (No. 5): pp. 787–800, 2000.

[Hallendorn, 1996a] Palm R. Driankov D. Hallendorn, H. *"Model Based Fuzzy Control"*. Springer Verlag, 1996.

[Hallendorn, 1996b] Palm R. Driankov D. Hallendorn, H. A takagi-sugeno fuzzy gain scheduler. 1996.

[Houghton, E.L. and Brock, A.E., 1970] Houghton, E.L. and Brock, A.E. *"Aerodynamics for Engineering Students"*. Arnold Publishing, 1970.

[Johansen, 1993] Foss B. A. Johansen, T. Constructing marmax models using armax models. *International Journal of Control*, 58(5):1125–1153, 1993.

[K. R. Lee and Al., 2000] K. R. Lee and J. H. Kim Et. Al. "Output Feedback Robust Control of Uncertain Fuzzy Dynamic Systems with Time Varing Delay". *IEEE Transaction on Fuzzy Systems*, 8(6):657–664, December 2000.

[Khargonekar, 1987] Zhou K. Khargonekar, P. Stability robustness bounds for linear state-space models with structured uncertainty. *IEEE Transactions on AC*, 32(7):621–632, July 1987.

[M. Pachter and J.L. Dargan, 1994] J. J. D'Azzo M. Pachter and J.L. Dargan. "Automatic Formation Flight Control". *Journal of Guidance, Control and Dynamics*, 17(6):1380–1383, May 1994.

[M. Pachter *et al.*, 2001] M. Pachter, J. J. D'Azzo, and A. W. Proud. "Tight Formation Flight Control". *Journal of Guidance, Control and Dynamics*, 24(2):246–254, March-April 2001.

[Marullo, 2001] A. Marullo. *Controllo Fuzzy-Gain Scheduling Per Sistemi Dinamici.* PhD thesis, Università di Pisa, 2001.

[McLain, T. W. *et al.*, 2000] McLain, T. W., Chandler, P., and Patcher, M. "A Decomposition Strategy for Optimal Coordination of Unmanned Air Vehicles". In *Proceedings of American Control Conference*, Chicago, IL, June 2000.

[Mesbahi, M. and Hadaegh, F. Y., 1999] Mesbahi, M. and Hadaegh, F. Y. "Formation Flight Control of Multiple Spacecraft: Graph Theoretic Properties and Switching Scheme". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, Portland, OR, August 1999.

[Meshabi, M. and Hadaegh, F. Y., 2001] Meshabi, M. and Hadaegh, F. Y. "Formation Flying Control of Multiple Spacecraft via Graphs, Matrix Inequalities, and Switching". *AIAA Journal of Guidance, Control, and Dynamics*, Vol. 24 (No. 3): pp. 369–377, 2001.

[Myatt, J. H. and Blake, W., 1999] Myatt, J. H. and Blake, W. "Aerodynamics Database Issues for Modeling Close Formation Flight". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, Portland, OR, August 1999.

[Nelson, R. C., 2001] Nelson, R. C. "Aircraft Wake vortices and their Effect on Following Aircraft". In *Proceedings of AIAA Atmospheric Flight Mechanics Conference*, Montreal, Quebec, Canada, 2001.

[Nichols, 1993] Reichert R. T. Rugh W. Nichols, R. A. Gain scheduling for hinf controllers: A flight control example. *IEEE Transactions on CST*, 1(2):69–79, February 1993.

[Proud. A. W. *et al.*, 1999] Proud. A. W., Pachter, M., and D'Azzo, J. J. "Close Formation Flight Control". In *Proceedings of AIAA Guidance, Navigation and Control Conference*, Portland, OR, August 1999.

[Schumacher and Kumar, 2000] C. Schumacher and R. Kumar. "Adaptive Control of UAVs in Close Formation Flight". In *Proceedings of American Control Conference*, Chicago, IL, June 2000.

[Takagi, 1985] Sugeno M. Takagi, T. Fuzzy identification of systems and its application to modeling and control. *IEEE Transactions on SMC*, 15(2):116–132, 1985.

[Tyne and Berger, 1976] J. Van Tyne and A.J. Berger. *"Fundamentals of Ornithology"*. John Wiley, 1976.